# LANDSLIDE INFORMATION SERVICE BASED ON COMPOSITION OF PHYSICAL AND SOCIAL INFORMATION SERVICES

A Dissertation
Presented to
The Academic Faculty

by

Aibek Musaev

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Computer Science

Georgia Institute of Technology
August 2016

# LANDSLIDE INFORMATION SERVICE BASED ON COMPOSITION OF PHYSICAL AND SOCIAL INFORMATION SERVICES

Approved by:


Professor Dr. Calton Pu, Advisor
School of Computer Science
at the College of Computing
*Georgia Institute of Technology*

Professor Dr. Edward R. Omiecinski
School of Computer Science
at the College of Computing
*Georgia Institute of Technology*

Professor Dr. Ling Liu
School of Computer Science
at the College of Computing
*Georgia Institute of Technology*

Professor Dr. Qingyang Wang
Division of Computer Science and Engineering
at the College of Engineering
*Louisiana State University*

Professor Dr. Shamkant B. Navathe
School of Computer Science
at the College of Computing
*Georgia Institute of Technology*


Date Approved: May 2, 2016

*To my beloved family, relatives and friends.*

# ACKNOWLEDGEMENTS

My dissertation would not have been possible without the support of so many people in my life. It was a long and winding road to my dissertation defense, and now it is time to express my gratitude to those who helped me so much along the way.

First and foremost, I would like to acknowledge the support of Prof. Calton Pu, who was a patient teacher and understanding advisor. I want to thank each member of my dissertation committee for their help and invaluable advices, namely Prof. Ling Liu, Prof. Shamkant B. Navathe, Prof. Edward Omiecinski, and Prof. Qingyang Wang. I also want to thank my former Prof. Mark Guzdial who introduced me to my advisor.

I am grateful to my colleagues from Georgia Tech. Special thanks to De Wang, Junhee Park, Qi Zhang, Chien-An Lai, Jack Li, Yuzhe Tang, Tao Zhu, and Chien-An Cho. De is a valuable colleague to work with, who helped me so much in the beginning of my research career. Junhee Park, Qi Zhang, Chien-An, Jack Li, Yuzhe Tang, Tao Zhu, and Chien-An Cho are great lab mates.

Most importantly, I am in forever debt to my wife and son for your love, courage and sacrifice. You have supported me through the ups and downs in this journey and I will always remember that. No words can describe my gratitude and love to you. I am forever grateful to my parents who gave me the gift of life, always believed in me and supported us through this journey. I am so much grateful to my second parents for your love and support. I also thank each member of my big family - your belief in and love for me is what kept me going. Finally, I want to thank all of my friends who helped us, including Kanybek and Munara Nur-tegin, Bolot Kerimbaev, Valery and Valentina Seleznev.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# SUMMARY

Modern world data come from an increasing number of sources, including data from physical sources like satellites and seismic sensors as well as social networks and web logs. While progress has been made in the filtering of individual social networks, there are significant advantages in the integration of big data from multiple sources. For physical events, the integration of physical sensors and social network data can improve filtering efficiency and quality of results beyond what is feasible in each individual data stream. Disasters are representative physical events with real world impact. In this dissertation, I present the LITMUS system that combines data from both physical sensors and social networks to provide information about physical events in near real-time.

My work consists of four parts: 1) integration of multiple sources for landslide detection, 2) filtering out noise from social media, 3) geo-tagging data from social media, and 4) sharing collected data with research community. In part I, I introduce the physical event information service LITMUS, which combines multiple physical sensors and social media to handle the inherent varied origins and composition of multi-hazards, such as landslides. In part II, I propose a classification approach based on similarity of texts to Wikipedia articles followed by a new approach for fast text classification using randomized ESA, and further improve classification accuracy using a rapid ensemble classification system. In part III, I address the challenge of lack of geo-tagged data in social media by proposing location estimation based on a composition of clustering algorithms. In part IV, I describe our Twitter dataset of landslide events and illustrate its uses. This is one of the largest annotated datasets available to date.

# CHAPTER I

# INTRODUCTION

The world is experiencing rapid growth in the amount of published data, which comes from various sources, including physical sensors and social networks. While progress has been made in the analysis of individual social networks, e.g., trend analysis in Twitter [35], there are significant advantages in the integration of big data from multiple sources. For physical events, the integration of physical sensors and social network data can improve filtering efficiency and quality of results beyond what is feasible in each individual data stream. Disasters are representative physical events with real world impact.

Natural disaster detection and management is a significant and non-trivial problem, which has been studied by many researchers. A conventional approach relies on dedicated physical sensors to detect specific disasters, e.g., using real-time seismometer data for post-earthquake emergency response and early warning [28]. A more recent approach explores the big data from social networks such as Twitter functioning as social sensors [49]. Since physical sensors (e.g., seismometers) are specialized for specific disasters, people have placed high expectations on social sensors. Besides, few physical sensors exist for the detection of multi-hazards such as landslides, which have multiple causes (earthquakes and rainstorms, among others) and happen in a chain of events. However, despite some initial successes, social sensors have met serious limitations due to the big noise in the big data generated by social sensors. For example, Twitter filter for the word "landslide" gets more tweets on the 70's rock song "Landslide" than landslide disasters that involve soil movement. News channels provide reliable and mostly verified information sources. Unfortunately, they normally

have high latency that may be up to several days after the occurrence of a disaster.

Besides, disasters like multi-hazards present more significant challenges, since there are no effective physical sensors that would detect multi-hazards directly. Landslide, which can be caused by earthquakes, rainfalls and human activity among other reasons, is an illustrative example of a multi-hazard. After investigating existing approaches using physical and social sensors, we propose a new physical event information service – LITMUS and also implement a prototype system in practice, which is based on a multi-service composition approach to the detection of landslides. More concretely, LITMUS has the following benefits compared with traditional or existing approaches for natural disaster detection:

- It composes information from a variety of sensor networks including both physical sensors (e.g., seismometers for earthquakes and weather satellites for rainfalls) and social sensors (e.g., Twitter and YouTube). Besides providing wider coverage than a system relying on a single source, it improves detection accuracy and reduces overall latency.

- It applies state-of-art filters for each social sensor and then adopts geo-tagging to integrate the reported events from all physical and social sensors that refer to the same geo-location. Such integration achieves better landslide detection when compared to an authoritative source. Meanwhile, the geo-location information not only provides the base for the integration, but also enables us to do real-time notification in the future.

- It provides a generic approach to the composition of multiple heterogeneous information services and uses landslide detection as an illustrative example, i.e. it is not tied to disaster detection and can be applied to other application areas involving service composition. Traditional approach to the composition of web services makes strong assumptions about services, which it then uses to select

services when composing a new service, such as quality of service [48] or service license compatibility [15]. In practice, the real world services do not satisfy such assumptions. The claim we make is that more information services should provide a more solid result and we demonstrate that it is the case with LITMUS.

In this thesis, we describe LITMUS, which filters and combines reliable but indirect physical data (e.g., rainfalls and earthquakes) with direct report (but noisy) social media data on landslides to achieve high quality and wide coverage of landslide information. Techniques that contribute to the high quality results include clustering approaches to geo-tagging and ensemble classification to noise filtering. We illustrate the described techniques using a set of data from Social Media that covers the whole year of 2014. This fully annotated and geotagged dataset contains 255k items, which we make openly accessible as a contribution to research community.

LITMUS software tools have been publicly released and a live demo runs on our project web portal. These software tools are being used in the extensions of LITMUS, both in depth (e.g., multi-language reporting of landslides worldwide) and in breadth (e.g., detection and monitoring of other disasters such as harmful algal blooms).

## 1.1 Dissertation Statement and Dissertation Contributions

Before proceeding to concrete contributions of my thesis, my thesis statement can be formulated as follows:

**Thesis Statement:** *Detection of natural disasters like landslides requires a composition of physical and social information services that can be effectively accomplished through real-time detection framework that combines data by filtering and then joining the information flow from those services based on their spatiotemporal features.*

To support my thesis statement, we make the following four contributions:

- We propose LITMUS (Section 2) — a landslide detection service based on a multi-service composition approach that combines data from both physical and

social information services by filtering and then joining the information flow from those services based on their spatiotemporal features. We evaluate the proposed approach using a real world dataset and compare the results of landslide detection by LITMUS versus an authoritative source.

- LITMUS uses a number of keywords to extract the data from Social Media related to landslides, including *landslide* and *mudslide*. However, most of the data returned by social networks are irrelevant to landslide as a disaster, where *landslide* is frequently used as an adjective describing an overwhelming majority of votes or victory and *mudslide* is often used as a popular cocktail. We convert the filtering problem to binary classification problem by considering relevant and irrelevant items as two classes. We implement multiple classification approaches in LITMUS and begin with a classification algorithm based on similarity of Social Media texts to Wikipedia articles describing relevant and irrelevant landslide concepts (Section 3). Next we resolve the sense of ambiguous search keywords (Section 4) using a reduced Explicit Semantic Analysis (ESA) approach. In particular, we propose RS-ESA method based on a random sample of Wikipedia articles used as a knowledge repository by ESA and Expert-ESA method based on a subset of Wikipedia articles selected using an expert driven approach. Finally, we improve the quality of the filtering process with a rapid ensemble classification system REX (Section 5), which outperforms the standard Bag-of-Words algorithm by an average of 0.14 and the state-of-the-art Word2Vec algorithm by 0.04.

- We are only interested in geo-tagged data as each disaster event has a point in time and space, however majority of data returned by social networks is not geo-tagged. We evaluate several approaches that retrieve geographic locations based on the mentions of place names that refer to locations of landslides

in the item's text (Section 6.1). We find that the named entity recognition (NER) based approach produces the least number of irrelevant locations and has the best precision and recall for geo-tagging purposes among the evaluated approaches. We next discuss the need for revision of cell-based integration (Section 6.2). Finally, we improve the quality of the geo-tagging component in LITMUS (Section 6.3) by proposing a clustering composition approach. In particular, location outliers are removed using clustering based on semantic distance, which is followed by clustering based on Euclidean distance, such that locations that are in close proximity to one another are grouped into the same cluster. Based on our knowledge, this is the first work that employs a composition of clustering algorithms to accurately estimate geographic locations based on unstructured texts.

- Over the course of our project we collected and processed a large amount of data from Social Media with respect to landslides. We share our annotated dataset and demonstrate how to apply the described techniques to it. The dataset covers the full year of 2014 and contains 255k items from Twitter, which makes it one of the largest annotated datasets to date. We describe the data collection, annotation and geotagging processes, and provide several experiments illustrating its usage.

## 1.2 Organization of the Dissertation

We split each contribution into a separate part, which contains chapters that illustrate the details of research work, to emphasize the components of a landslide information system that combines data from both physical sensors and social networks in near real-time. We attempt to keep each chapter an independent unit with its own evaluation and related work.

**Part I. Integration of Multiple Sources for Landslide Detection**

- **Chapter 2: LITMUS: A Landslide Detection Service based on Multiple Sources**. Disasters often lead to other kinds of disasters, forming multi-hazards such as landslides, which may be caused by earthquakes, rainfalls, water erosion, among other reasons. Effective detection and management of multi-hazards cannot rely only on one information source. In this chapter, we evaluate a landslide detection system LITMUS, which combines multiple physical sensors and social media to handle the inherent varied origins and composition of multi-hazards. LITMUS integrates near real-time data from USGS seismic network, NASA TRMM rainfall network, Twitter, YouTube, and Instagram. The landslide detection process consists of several stages of social media filtering and integration with physical sensor data, with a final ranking of relevance by integrated signal strength. Applying LITMUS to data collected in October 2013, we analyzed and filtered 34.5k tweets, 2.5k video descriptions and 1.6k image captions containing landslide keywords followed by integration with physical sources based on a Bayesian model strategy. It resulted in detection of all 11 landslides reported by USGS and 31 more landslides unreported by USGS. An illustrative example is provided to demonstrate how LITMUS' functionality can be used to determine landslides related to the recent Typhoon Haiyan.

**Part II. Filtering Out Noise from Social Media**

- **Chapter 3: Classification Approach based on Similarity of Texts to Wikipedia Articles**. In this chapter, we describe and evaluate a prototype implementation of a landslide detection system called LITMUS, which combines multiple physical sensors and Social Media to handle the inherent varied origins and composition of multi-hazards. The landslide detection process consists of several stages of Social Media filtering and integration with physical sensor data, with a final ranking of relevance by integrated signal strength. The filtering component of the prototype uses a classification algorithm that determines

the relevance of Social Media items based on their similarity to relevant and irrelevant Wikipedia articles. The chapter describes the process of how the corresponding Wikipedia articles are collected and the use of Jaccard distance as the similarity measure. Our results demonstrate that with such approach LITMUS detects 41 out of 45 reported events as well as 165 events that were unreported by the authoritative source during the evaluation period.

- **Chapter 4: Fast Text Classification Using Randomized Explicit Semantic Analysis**. Document classification or document categorization is one of the most studied areas in computer science due to its importance. The problem is to assign a document using its text to one or more classes or categories from a predefined set. We propose a new approach for fast text classification using randomized explicit semantic analysis (RS-ESA). It is based on a state-of-the-art approach for word sense disambiguation based on Wikipedia, the largest encyclopedia in existence. Our method reduces Wikipedia repository using a random sample approach resulting in a throughput, which is an order of magnitude faster than the original explicit semantic analysis. RS-ESA approach has been implemented as part of the LITMUS project due to the need for classifying data from Social Media into relevant and irrelevant items with respect to landslide as a natural disaster. We demonstrate that our approach achieves 96% precision when classifying Social Media landslide data collected in December 2014. We also demonstrate the genericity of the proposed approach by using it to separate factual texts from fictional based on Wikipedia articles and fan fiction stories, where we achieve 97% in precision.

- **Chapter 5: REX: Rapid Ensemble Classification System for Landslide Detection using Social Media**. We study the problem of using Social Media to detect natural disasters, of which we are interested in a special kind, namely

landslides. Employing information from Social Media presents unique research challenges, as there exists a considerable amount of noise due to multiple meanings of the word "landslide". To tackle these challenges, we propose REX, a rapid ensemble classification system which can filter out noisy information by implementing two key ideas: (I) a new method for constructing independent classifiers that can be used for rapid ensemble classification of Social Media texts, where each classifier is built using randomized Explicit Semantic Analysis; and (II) a self-correction approach which takes advantage of the observation that the majority label assigned to Social Media texts belonging to a large event is highly accurate. We perform experiments using real data from Twitter over 1.5 years to show that REX classification achieves 0.98 in F-measure, which outperforms the standard Bag-of-Words algorithm by an average of 0.14 and the state-of-the-art Word2Vec algorithm by 0.04. We also release the annotated datasets used in the experiments as a contribution to the research community containing 282k labeled items.

**Part III. Geo-tagging Data from Social Media**

- **Chapter 6: Location Estimation based on Clustering Composition**. The use of Social Media for event detection, such as detection of natural disasters, has gained a booming interest from research community as Social Media has become an immensely important source of real-time information. However, it poses a number of challenges with respect to high volume, noisy information and lack of geo-tagged data. Extraction of high quality information (e.g., accurate locations of events) while maintaining good performance (e.g., low latency) are the major problems. In this chapter, we propose a composition of clustering algorithms for location estimation. Our experiments demonstrate a 20% improvement in location estimation due to clustering composition approach. We implement this approach as part of the landslide detection service LITMUS,

which is live and openly accessible for continued evaluation and use.

**Part IV. Sharing Collected Data with Research Community**

- **Chapter 7: Annotated Dataset of Landslide Events from Twitter**. We introduce the annotated dataset of landslide events from Twitter. The dataset covers the full year of 2014 and the keywords used to collect it include "landslide" and "mudslide". The tweets are annotated based on their relevance to landslide as a natural disaster, and the events are defined based on their spatiotemporal features. In this chapter, we describe the data collection process and the annotation of data, and also explain the geotagging process. To date, this is the most comprehensive research dataset dedicated to a particular type of disaster events. It is also one of the largest annotated datasets. We provide several illustrations of its possible uses, including visualization of landslide activity based on retweets, evaluation of classification performance and detection of landslide events.

**Chapter 8: Conclusion**. In this chapter, we wrap up the dissertation.

# CHAPTER II

# LITMUS: A LANDSLIDE DETECTION SERVICE BASED ON MULTIPLE SOURCES

## 2.1 Introduction

In this chapter, we describe and evaluate a landslide detection system LITMUS, which is based on a multi-source integration approach to the detection of landslides, a representative multi-hazard. LITMUS integrates information from a variety of sensor sources instead of trying to refine the precision and accuracy of event detection in each source. Our sources include both physical sensors (e.g., seismometers for earthquakes and weather satellites for rainstorms) and social sensors (e.g., Twitter and YouTube). Although we still have some technical difficulties with filtering out noise from each social sensor source, LITMUS performs a series of filtering steps for each social sensor, and then adopts geo-tagging to integrate the reported events from all physical and social sensors that refer to the same geo-location. Our evaluation shows that with such integration the system achieves better precision and F-measure in landslide detection when compared to individual social or physical sensors.

This chapter makes several contributions. The first contribution is the construction of a landslide detection system LITMUS that integrates online feeds from five sources. Two of sources are physical sensors: seismic activity feed provided by USGS and rainfall activity feed provided by NASA TRMM. Three sources are social sensors: Twitter for text information, Instagram for photos, and YouTube for videos. We believe the combination of these relatively independent sources of data enables LITMUS to improve the precision and accuracy of landslide detection. The second contribution is a quantitative evaluation of the system using real world data collected

in October 2013. LITMUS detected all 11 landslides reported by USGS as well as 31 more landslides unreported by USGS during this period. The final contribution is an illustrative example of the functionality of the system to determine a list of landslides caused by the recent Typhoon Haiyan, which devastated the Philippines on November 8th.

## 2.2 Overview of Approach

For better understanding of our landslide detection system LITMUS, we present an overview of the system's data flow in Figure 1.

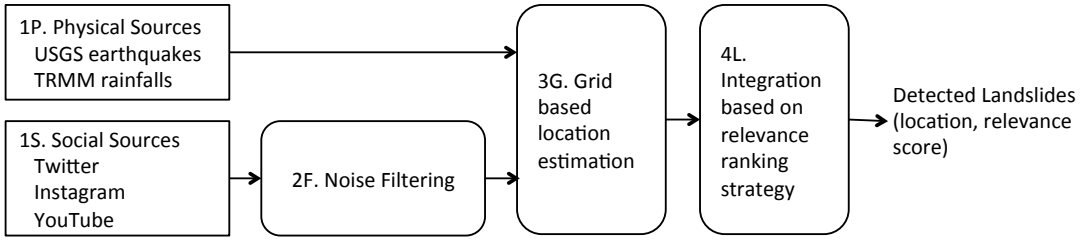


**Figure 1:** Overview of LITMUS framework

The system starts with the raw data collection. It periodically downloads data from multiple social and physical sensors. The social sensors supported by LITMUS are popular social network sites, namely Twitter, YouTube, and Instagram. Each of these sensors is among the leading social networks in their respective areas. LITMUS extracts the data from these sensors by applying a search filter based on landslide related keywords. We perform noise filtering in a series of filtering steps, including filtering out items based on stop words and stop phrases, filtering items with accurate geo-tags based on the geo-tagging component, filtering relevant items based on the machine learning classification component, and filtering out items based on a blacklist of URLs – see Figure 2, where "+" indicates an inclusion type of filtering and "-" indicates an exclusion type of filtering. LITMUS also collects data from the physical sensors, namely the seismic activity and the rainfall activity feeds. We support them in our system because these feeds are related to hazards that may cause landslides.

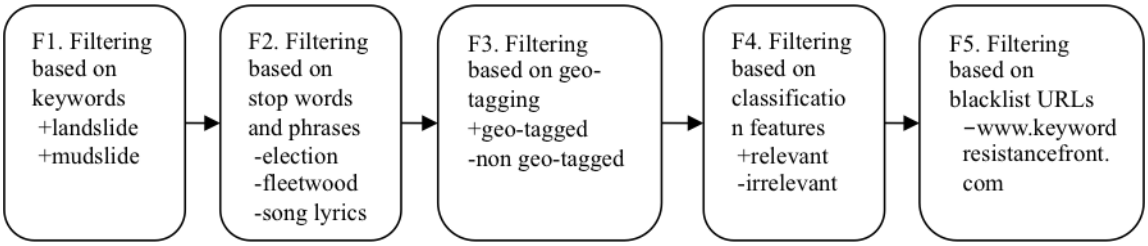

**Figure 2:** Overview of filtering pipeline

In the end, we combine the remaining items from the social sensors with all items from the physical sensors based on the relevance ranking integration strategy. The final output of the system is a list of detected landslides with location information and relevance scores.

## 2.3 1P. Physical Sources Support

LITMUS collects data from several physical sensors. In particular, LITMUS supports a real-time seismic activity feed from the United States Geological Survey (USGS) agency[1]. This feed is updated every minute providing information about earthquakes of various magnitudes. LITMUS downloads the data from USGS on earthquakes of 2.5 magnitude and higher. USGS provides programmatic access via a well-structured GeoJSON format that can be conveniently parsed. It provides time, magnitude, latitude, longitude, name of the place where an earthquake occurred and an event ID.

Another potential cause of landslides is rainfalls, which is why LITMUS also collects data from the Tropical Rainfall Measuring Mission (TRMM) project[2]. It is a joint project between NASA and the Japan Aerospace Exploration Agency (JAXA), which generates reports based on the satellite data of the areas on the planet that have experienced rainfalls within the past one, three and seven days. The reports are in multiple formats, including a web page on the project's portal, from which

[1]http://earthquake.usgs.gov/earthquakes/feed/v1.0/geojson.php
[2]http://trmm.gsfc.nasa.gov/

LITMUS periodically downloads and parses data about rainfalls.

## 2.4 1S. Social Sources Support

### 2.4.1 F1. Filtering based on keywords

According to the research on citizen activity [44], social networks have emerged as destinations for collective disaster-related sensemaking. LITMUS uses the data from social networks to help detect landslides as reported by the public. In particular, LITMUS downloads the data from Twitter as an example of a text based social network, YouTube as an example of a video based social network and Instagram as an example of an image based social network. All listed social networks provide programmatic access to their data via search API based on keywords. LITMUS downloads the data from each social network based on "landslide" and "mudslide" keywords.

### 2.4.2 F2. Filtering out based on stop words and stop phrases

Next, LITMUS performs filtering by excluding social sensor items that contain negative stop words with respect to landslides, such as "fleetwood" or "election". The following is a set of examples from Twitter that represent unrelated to landslides items containing these stop words:

- *"Landslide by Fleetwood Mac will forever be one of my favorite songs."*
- *"Abbott builds on election landslide: TONY Abbott is riding a post-election honeymoon high, with nearly half of... http://t.co/P17WAyxud2"*

LITMUS also removes items based on stop phrases that currently contain excerpts from the lyrics of some popular songs that are commonly used in social networks, e.g. the lyrics from the "Landslide" song by Stevie Nicks from Fleetwood Mac: *"...and I saw my reflection in the snow covered hills..."*

### 2.4.3 F3. Filtering based on geo-tagging

After LITMUS downloads the data from the physical and social sensors, we need to obtain geo locations of the downloaded data. The data from the physical sensors already contains geo coordinates. Unfortunately, the data from the social sensors is usually not geo-tagged since few users disclose their locations. Thus, if an item coming from a social source is not geo-tagged, we need to look for geo terms inside the textual description of the item. An important component included in social sensor items is mentions of place names that refer to locations of landslides. An exact match of words in the textual description of an item is performed against the list of all geo terms. For the list of geo terms we use the approach described in [22] to locate accurate geo coordinates based on the titles of the geo-tagged Wikipedia articles. However, different types of geo coordinates are supported in the geo-tagged Wikipedia articles. Some of them, like "city" or "country", are more relevant than others, such as "landmark", which often returns irrelevant matches like "houses" or "will".

However, the relevance quality of this algorithm should be improved further. For example, some geo terms may appear valid, such as "Says", which was a municipality in Switzerland, or "Goes", which is a city in Netherlands, however they are also verbs that are commonly used in English texts. That is why prior to applying the geo-tagging algorithm on the downloaded social media data, LITMUS performs pre-filtering of the words inside those items using Part-Of-Speech (POS) tagging by excluding non-noun words from consideration.

There are also geo terms like "cliff" or "enterprise" whose type is "city" that are not very helpful for the purpose of landslide location estimation. The algorithm would incorrectly retrieve "cliff" as a geo term from the following YouTube item: "Driver Survives Insane Cliff Side Crash." The reason why these words are irrelevant is because they happen to be common nouns, in other words they are used in English

texts a lot. To mitigate this issue we use a list of 5000 most frequent words in English based on the Corpus of Contemporary American English[3] and exclude those results from the list of geo terms.

Among the supported social sensors, YouTube in particular contains a lot of items where in addition to some valuable information related to landslides, they also contain unrelated information. The following is an illustrative example that follows such pattern:

- *"After fatal Flash Flood, Mudslide, More Rain Possible for Colorado and other states youtube original. news bloopers, fox news,onion news,funny news bloopers, news failbreaking news,bbc news news reporter news fails cbs news cnn news world news us news uk news syria today syria war syria 2013 syria new,syria news,damascu,syria damascus, syrian army,syrian,syria execution..."*

It is clear that "Colorado" is a relevant geo term, whereas "Syria" and "Damascus" are not. In order to take into account patterns like this, we augmented the geo-tagging algorithm as follows: the input text is broken into sentences and for each sentence we find the geo term that is the closest to the landslide keyword. In this example the landslide keyword is "mudslide" and the closest available geo term is "Colorado", hence the geo-tagging algorithm correctly outputs "Colorado".

### 2.4.4 F4. Filtering based on machine learning classification

The social sources in LITMUS frequently return items that are not relevant to landslides, even though they contain landslide keywords. The following is an example of irrelevant items that use "landslide" as an adjective describing an overwhelming majority of votes or victory: *"We did it! Angel won in Starmometer 100 Most Beautiful Women in the Philippines for 2013! Landslide victory due... http://t.co/2g6ozhJhpj"*

[3]http://www.wordfrequency.info

To filter out such items from the social sensors LITMUS employs binary classification, a machine learning technique to automatically label each item as either relevant or irrelevant based on classifier model built from a training set containing labeled items. To prepare a training set we need a list of confirmed landslides. For this purpose we use expert landslide publications. The USGS agency, in addition to earthquakes, also publishes a list of landslide events collected from external reputable news sources, such as Washington Post, China Daily, Japan Times and Weather.com[4]. For each event in this list we identify the date of release and geo terms.

To find the social network items related to confirmed landslides within each month, we first filtered the data based on the landslide locations extracted from the confirmed landslides. Then we manually went through each item in the filtered list to make sure that they described corresponding landslides by comparing the contents of the items with the corresponding landslide articles. And whenever there were URLs inside those social items, we looked at them also to make sure that they were referring to the corresponding landslides.

The following is an example of a landslide confirmed by the Latin Times news source, which was published on September 11, 2013: *"Mexico Mudslide 2013: 13 Killed in Veracruz Following Heavy Rains."* The geo terms that LITMUS extracted from this news title are "Mexico" and "Veracruz".

To create a list of unrelated items in the training set, we randomly picked items from each social source and manually went through each item. But this time we had to make sure that the items did not describe landslide events.

### 2.4.5 F5. Filtering based on blacklist URLs

During the analysis of social media items containing URLs, we found out that in several cases the short URLs were expanded into the same web site[5] that generated

[4]http://landslides.usgs.gov/recent/
[5]http://keywordresistancefront.com

random content with high-value keywords such as "mudslide". Based on this result we created a blacklist of URLs and added a filter to exclude items containing such URLs from consideration.

## 2.5 3G. Grid based location estimation

As a result of the previous stages in the system's data flow shown in Figure 1, LITMUS has a set of relevant and geo-tagged items from physical and social sensors. Next LITMUS integrates those items by grouping them based on their geo coordinates to determine areas on the planet where landslides might have occurred. For this purpose we propose to represent the surface of the Earth as a grid of cells. Each geo-tagged and relevant to landslide item is mapped to a cell in this grid based on the item's geo coordinates. After all items are mapped to cells in this grid, the items in each non-empty cell are counted per each source. Currently we use a 2.5 minute grid both in latitude and longitude, which corresponds to the resolution of the Global Landslide Hazard Distribution. This is the maximum resolution allowed by the system. The actual resolution is driven by the precision of the geo-tagging algorithm described in Section F3.

## 2.6 4I. Integration based on relevance ranking strategy

After mapping all items to cells in the grid, we obtain a set of non-empty cells. These cells represent areas on the planet where landslides may have occurred. To tell which cells are more likely to have experienced landslides, we propose a Bayesian model strategy and compare it with two baseline strategies – "OR" and "social AND physical". For "OR" integration strategy, we grant equal weights to all sensors. And we obtain the decision by combining the votes using boolean operation OR among five sensors. For "social AND physical" integration strategy, we use boolean operation OR to combine the votes from social sensors and physical sensors separately first. And then we calculate the combined result by applying boolean operation AND between

votes from social and physical sensors. For instance, if the votes from five sources (Twitter, Instagram, YouTube, USGS, and TRMM) are 1,1,0,0, and 0, the "OR" strategy will return 1, but the "social AND physical" strategy will return 0.

The description of the Bayesian model strategy is as follows. Suppose, there is a cell $x$ and $\omega$ is the class associated with $x$, either being true or false. Then, assuming a hidden variable $Z$ for an event to select one source, a probability for a class $\omega$ given $x$, $P(\omega|x)$, can be expressed as a marginal probability of a joint probability of $Z$ and $\omega$:

$$P(\omega|x) = \sum P(\omega, |Z_i|x) = \sum_i P(\omega|Z_i, x)P(Z_i|x), \tag{1}$$

where external knowledge $P(Z_i|x)$ denotes each source's confidence given $x$. For instance, if a certain source becomes unavailable, the corresponding $P(Z_i|x)$ will be zero. Also one could assign a large probability for the corresponding $P(Z_i|x)$ if one source dominates over other sources.

In our experiment, to provide a balance between precision and recall, we use prior F-measure $C$ from the training dataset as the confidence for each source. Keeping the results in the range from 0 to 1, we normalize the values of F-measure into a scale between 0 and 1 first. After taking the number of items $N$ from each source into account, the formula will be further converted into the following format:

$$P(\omega|x) = \sum C_i \frac{N_i^x}{N_i^x + 1}, \tag{2}$$

where $C_i$ denotes the normalized prior F-measure of source $i$ from historic data (we use August and September data in our experiments). $N_i^x$ denotes the number of items from source $i$ in cell $x$ indicating that a landslide occurred in the area covered by cell $x$. It should be noted that for Bayesian model strategy we ignore cells with only 1 vote, i.e. where the total count of items in that cell is equal to 1. This is done to reflect the idea of a multi-source integration as opposed to a single source analysis.

## 2.7 Implementation Summary

LITMUS is developed using free and open-source software. It consists of a front-end implemented as a Web application and a back-end, which is the core of the system. The front-end is a live demonstration that runs on Apache web server. It uses Google Maps JavaScript API to render all feeds, including detected landslides, and PHP to access LITMUS' back-end. The back-end is developed in Python, except for binary classification for which we used Weka's library implemented in Java [18]. All data from social and physical sensors is stored in MySQL. The data has been collected since August 2013 and takes up 1.7GB on disk. The total number of lines of code is 12k.

## 2.8 Experimental Evaluation

In this section, we present an experimental study using LITMUS. We designed 4 sets of experiments to evaluate its performance. We start by analyzing the effectiveness of the filtering techniques that are employed in social sensors to retrieve landslide relevant items. Next we compare the performance of physical sensors that monitor seismic and rainfall activities as possible causes of landslides. In the third experiment we measure the effectiveness of 3 integration strategies of both social and physical sensors to find the optimal integration strategy. And in the last experiment we compare the overall performance of LITMUS in landslide detection using the chosen strategy versus an authoritative source of landslide events compiled by USGS.

### 2.8.1 Retrieval of Landslide Relevant Items from Social Media

*Overview of filtering results*

As we mentioned earlier, LITMUS performs a series of filtering steps on the data from social sources to retrieve landslide relevant items. Table 1 contains the results of the filtering steps on the data collected during the evaluation period, which is the

month of October in 2013. It shows that Twitter has the most number of items and that the geo-tagging component filters out most of items.

**Table 1:** Overview of filtering results in December 2013

| **Sources** | **F1.Filter based on keywords** | **F2.Filter based on stop words & phrases** | **F3.Filter based on** geo-tagging | **F4.Filter based on classification** | **F5. Filter based on blacklist URLs** | **4I. Integration based on relevance ranking strategy** |
|---|---|---|---|---|---|---|
| Twitter | 34508 | 24898 | 6107 | 4630 | 4624 | 3861 |
| Instagram | 1631 | 1403 | 178 | 13 | 13 | 8 |
| YouTube | 2534 | 2221 | 331 | 182 | 182 | 105 |

*Features used in classification*

The filtering step F4 employs SVM, which is an algorithm for training a support vector classifier. The training dataset needed by the algorithm consists of social source items in August and September 2013, including 12,328 tweets, 1,266 Instagram images and 3,174 YouTube videos. In classification, we extracted a set of features based on the textual description of items from each social source. In particular, we created 3 groups of features that are applied to each source:

1. Common statistical features: 1) length of the textual description, 2) number of uppercase characters, 3) position of the query term in the textual description divided by number of words, 4) number of lowercase characters, 5) total number of words, 6) maximum word length, 7) minimum word length, 8) average word length, 9) code of the most common character.

2. Binary features: presence of the following elements – 1) at sign, 2) URL, 3) percentage, 4) geo-term, 5) number, 6) hashtag, 7) exclamation mark, 8) question mark, 9) ellipsis, 10) double quotes, 11) colon, 12) heart symbol, 13) '♫' symbol.

3. Vocabulary based features: 1) relevant vocabulary score, 2) irrelevant vocabulary score. For these features we collect the lists of words (or vocabularies) based on the training set, which contains items labeled as relevant and irrelevant. For

each downloaded item we compute the total count of words that are present in the relevant vocabulary list, which we call a relevant vocabulary score, and also the total count of words that are present in the irrelevant vocabulary list, which we call an irrelevant vocabulary score.

The following is an example Tweet with the corresponding feature values below: *"Philippines - Travel News - Death toll reaches 32 following monsoon rains, landslides and flooding #Philippines #travel #safety #flooding"*

1. 1) 137, 2) 5, 3) 0.588, 4) 132, 5) 17, 6) 11, 7) 2, 8) 6.588, 9) 32.
2. 1) False, 2) False, 3) False, 4) True, 5) True, 6) True, 7) False, 8) False, 9) False, 10) False, 11) False, 12) False, 13) False.
3. 1) 0.185, 2) 0.099.

*Performance of social sensors*

To evaluate the performance of the social sensors we have used several criteria that are standard in the area of information retrieval, namely precision, recall and F-measure. Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. F-measure considers both precision and recall and is the harmonic mean of precision and recall.

Let us consider the relevance of the social sensors with respect to landslide disaster events based on these criteria. According to the results shown in Figure 3, Twitter has the highest recall as it has the most number of items among all sensors, whereas YouTube has the highest precision. Instagram showed the worst results among these sensors, as most of its images were unrelated to landslide events. Overall, Twitter has the highest F-measure in spite of its low precision, so any improvements in its precision should increase its F-measure even more.

*Analysis of Physical Sensors*

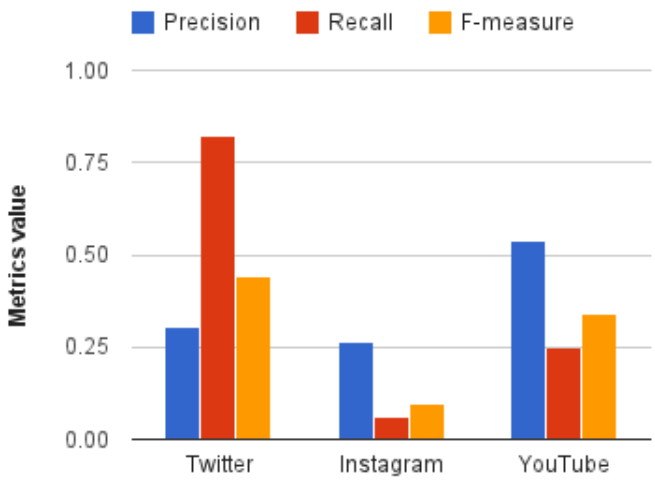


**Figure 3:** Landslide relevance of social sources

For our next experiment we compare the relevance of the physical sources with respect to landslide disaster events, namely the seismic and rainfall activities – see Figure 4. LITMUS collected 6,036 seismic activity points provided by USGS and 723 rainfall observations provided by TRMM in October. Due to such gap in the sheer volume of data, the seismic activity sensor shows better recall, but both precision and F-measure are better for rainfalls, which means that in October the influence of rainfalls on landslides was relatively higher.

### 2.8.2 Multi-Source Integration Strategies

In this experiment we compare the performance of the following relevance ranking strategies with respect to landslide detection: Bayesian model strategy versus two baselines – "OR" and "social AND physical" integration strategies shown in Figure 5. The "OR" strategy expectedly has the highest recall, because it includes all votes from each sensor in its decisions, which is also the reason why it has the lowest precision among all integration strategies. "Social AND physical" strategy produces a much better precision and F-measure, but very low recall. And the Bayesian model produces the best precision and F-measure results and an acceptable value of recall, which is

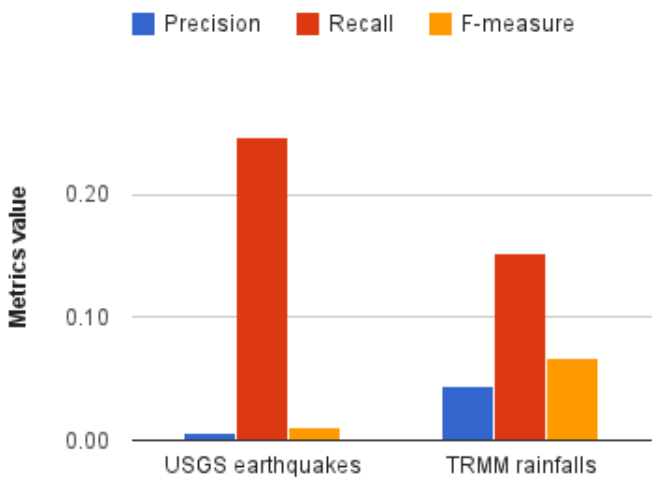


**Figure 4:** Landslide relevance of physical sources

why we select it as the best strategy for landslide detection among these strategies.

### 2.8.3 System Performance Results

LITMUS scripts run periodically where a period is customizable and currently set to 30 minutes. During each period, LITMUS performs a series of filtering steps F1 through F5 followed by integration step 4I. For each step we provide Latency and Throughput metrics to evaluate the system performance shown in Table 2.

**Table 2:** Overview of system performance results for a period from 2013-12-12 to 2013-12-19

| **Metrics** | **F1.Filter based on keywords** | **F2.Filter based on stop words & phrases** | **F3.Filter based on** geo-tagging | **F4.Filter based on classifi-cation** | **F5.Filter based on blacklist URLs** | **4I.Integration based on relevance ranking strategy** |
|---|---|---|---|---|---|---|
| Latency (s) | 1318.1 | 13.3 | 218.2 | 60.5 | 670.8 | 13.7 |
| Throughput (items/s) | 11.0 | 1090.2 | 35.8 | 37.9 | 1.2 | 462.5 |

Latency is a time interval between the beginning and end of each processing step and Throughput equals the total number of items processed at each step divided by the amount of time to process them.

F5 has the lowest throughput due to the cost of short URL expansion. F1 has low throughput due to pagination and extra delays when downloading the data. F3

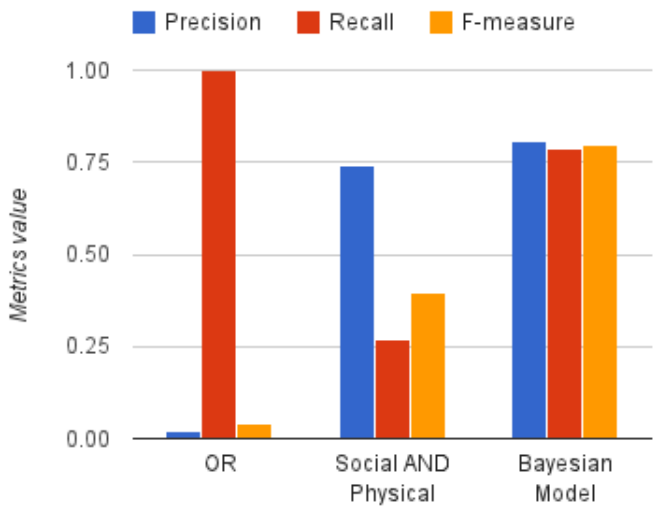


**Figure 5:** Landslide detection performance of integration strategies

and F4 also have low throughput due to the costs of geo-tag search and classification model generation.

### 2.8.4 Landslide Detection Results

LITMUS detected 42 landslide events in October. Of these 42, 11 were reported by USGS – see here[6]. In addition, LITMUS detected 31 landslides not reported by USGS. For each landslide we have performed manual verification by finding other reputable sources that would confirm the detected landslide events. We made sure that both locations and dates of the events were confirmed.

This is an example tweet regarding a landslide event that occurred in Obudu Resort, Nigeria in October, which was not reported by USGS: *"Over 20 People Trapped In Obudu Resort Mudslide http://t.co/aaOUn5465m"* (posted 10/17/2013). The tweet contains a shortened URL that points to a news article on Channels Television website[7], which confirms the location and the date of the landslide event.

It should be noted that 1 event reported by USGS was not an actual disaster

[6]http://landslides.usgs.gov/recent/index.php?year=2013&month=Oct

[7]http://www.channelstv.com/home/2013/10/17/over-20-people-trapped-in-obudu-resort-mudslide/?utm_source=dlvr.it&utm_medium=twitter

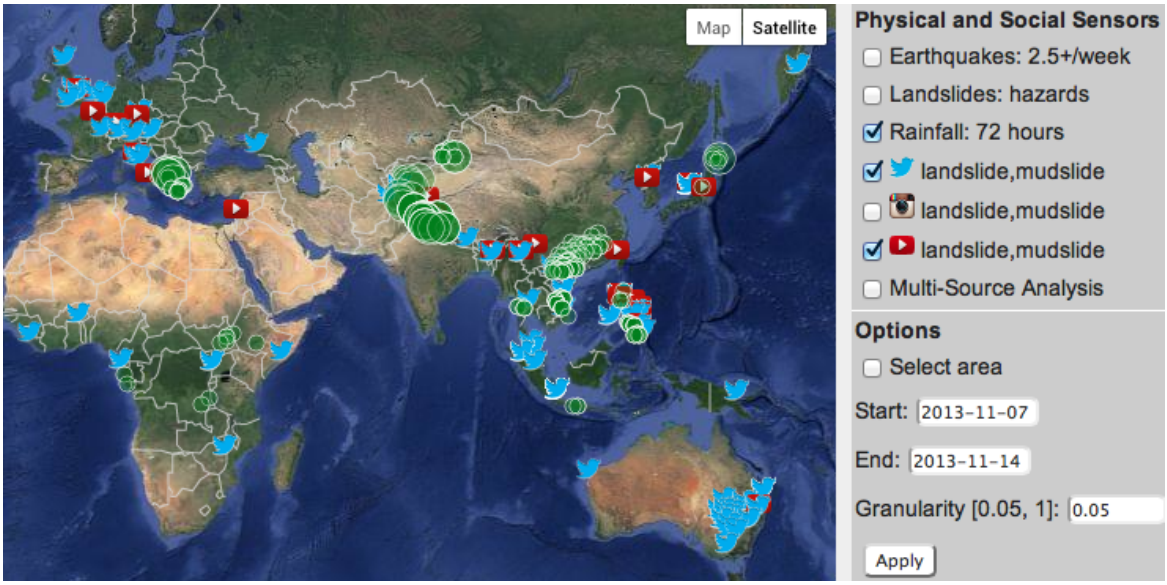


**Figure 6:** LITMUS live demonstration

report, namely: *"Flash Floods and Debris Flows: How to Manage Nature's Runaway Freight Trains"* (posted 10/30/2013). LITMUS successfully did not detect this report as a landslide event. All of the remaining reported events were successfully detected by the system.

## 2.9 Live Demonstration

We developed a live demonstration of the landslide detection system LITMUS as part of the GRAIT-DM project's web portal[8]. The web portal demonstrates multiple functionalities supported by LITMUS, including live feeds from each social and physical sensor, a separate feed of landslides detected by the system, support for viewing detailed information about each feed, and various user options to analyze results further – see Figure 6.

The data from all feeds is displayed on a Google Map. Each feed can be turned on and off to give a user an ability to view the data from a particular feed or a combination of feeds. Users can also obtain detailed information regarding each feed.

[8]https://grait-dm.gatech.edu/demo-multi-source-integration/

For example, if the feed is from the Instagram sensor, then they can view the related images. Similarly, if the feed is from the YouTube sensor then they can view the related videos. Finally, it should be noted that LITMUS has been collecting the data from all sensors since August 2013 only.

*Typhoon Haiyan (Yolanda)*

As an illustrative example of LITMUS functionality, let us consider the top event identified by the system in November 2013. As of November 14th, the table on the live demonstration page shows that within the last 7 days the cell with the top landslide score is the one for Philippines, which has been devastated by Typhoon Haiyan (known in the Philippines as Typhoon Yolanda) on November 8th. To find out which landslides have been caused by this event during this period we need to use the Select area option described above. Using this feature we cover the area of Philippines and recompute the results by applying the changes.

The table now shows 21 locations identified by LITMUS as landslide events in the selected area, including:

- Manila: *"100 dead as storm ripped apart buildings and triggered landslides in #Manila http://t.co/6tpLlxyBVy http://t.co/Sw09WyR6KQ"*
- Cebu: *"MT Province of Cebu @cebugovph 4m #YolandaPH Another landslide also reported in barangay Buhisan, #Cebu City #hmrd"*

1 result out of 21 was falsely identified as a landslide event, namely:

- Antipolo: *"No reported floods, landslides in #Antipolo as of 7:23 p.m. –Dodie Coronado, PIO — @KFMangunay @InqMetro #YolandaPH"*

## 2.10 Related Work

Disaster detection based on social media received a lot of attention in the last several years. Most of previous research studies focused on a single social network. For instance, [17] described Twitter Earthquake Detector (TED) system that infers the level of public interest in a particular earthquake based on Twitter activity to decide which earthquakes to disseminate to the public. [49] investigated the real-time nature of Twitter for detection of earthquakes. [4] compared different classification approaches on the Haiti disaster relief dataset obtained from the Ushahidi project. [54] investigated the performance of machine learning techniques in identifying on-the-ground twitterers during mass disruptions. [27] classified unstructured tweets into a set of classes and extracted short self-contained structured information for further analysis. Our disaster detection approach differs in several ways. We propose to integrate data from multiple social sources as opposed to a single social source. We also investigate the detection of multi-hazard disasters, in particular landslides, which can be caused by various hazards, such as earthquakes and torrential rains. That is why our system also integrates data from physical sources, including seismic activities and rainfalls.

Another important aspect of a disaster detection system based on social media is situational awareness. Although most of social networks provide support for users to disclose their locations, e.g. when they send a tweet or share a photo, [6] showed that less than 0.42% of all tweets actually use this functionality. [58] analyzed microblog posts to identify information that may contribute to enhancing situation awareness. [6] proposed and evaluated a probabilistic framework for estimating a Twitter user's city-level location based on the contents of tweets. [22] proposed to match locations in user profiles against the titles of Wikipedia articles containing geo coordinates. [21] showed that 34% of users did not provide real location information in their Twitter user profiles, and those that did input their locations – mostly specified at a city-level detail. [55] demonstrated a rapid unsupervised extraction of locations

references from tweets using an indexed gazetteer, which is a dictionary that maps places to geographic coordinates. Our system also extracts geo terms from the textual descriptions of data from social media using the Wikipedia articles containing geo coordinates as an indexed gazetteer. We improve the precision of this geo-tagging algorithm based on a number of heuristics to filter out irrelevant matches.

## 2.11 Conclusion

In this work we describe the landslide detection system LITMUS, which integrates multiple sources to detect landslides, a representative multi-hazard. In particular, the system integrates social sensors (Twitter, Instagram, and YouTube) and physical sensors (USGS seismometers and TRMM satellite). The data from social sensors is processed by LITMUS in a series of filtering steps, including data collection based on landslide keywords, a filter based on stop words and stop phrases, a smart geo-tagging filter, a machine learning based classification filter, and a filter based on a blacklist of URLs. The remaining data from social sensors as well as all data from physical sensors are combined for the final integration of all sensors to produce a list of detected landslides. The effectiveness of the system is evaluated using real world data collected in October 2013. The full integration of five sensor sources applying a modified Bayesian integration strategy detected all 11 landslides reported by USGS as well as 31 more landslides unreported by USGS during the evaluation period. The user functionality of the system as well as its application to Typhoon Haiyan is described in the Live Demonstration section. The landslide detection system LITMUS is online and openly accessible, collecting live data for continued evaluation and improvement of the system, and the reader is encouraged to use the demo.

# CHAPTER III

# CLASSIFICATION APPROACH BASED ON SIMILARITY OF TEXTS TO WIKIPEDIA ARTICLES

## 3.1 Introduction

Government through its agencies plays a critical role in disaster management. There are multiple government agencies dealing with various aspects of disasters, including FEMA and CDC. The Federal Emergency Agency (FEMA) is a federal agency under the Department of Homeland Security, which is responsible for coordinating the response to a disaster. The Centers for Disease Control and Prevention (CDC) is a federal agency under the Department of Health and Human Services. It is responsible for emergency preparedness and response. Unlike these two major agencies that are directly charged with handling disasters, the United States Geological Survey (USGS) is a scientific agency. It studies the landscape of the United States, its natural resources and the natural hazards that threaten it. But regardless of the type or purpose, all of these agencies utilize Social Media as part of their activities.

The agencies maintain a number of Social Media accounts as part of their mission to disseminate information to the public and even offer digital toolkits to integrate such information into third party tools[1]. USGS uses Social Media channels to inform the public about various natural hazards, including earthquakes, landslides and volcanoes[2]. However, Social Media itself can be used as a source of data for disaster management instead of solely relying on physical sensors. A good example of exploring the data from Social Media is Twitter data streams functioning as social sensors [49].

[1] http://www.cdc.gov/socialmedia/tools/guidelines/socialmediatoolkit.html
[2] https://twitter.com/usgsnewshazards

Also, many existing disaster management systems adopt multiple information sources, including news channels. However, they all face the challenge of integrating multiple information sources in the way that preserves the useful information while limiting the amount of noise. We cannot depend on a single information source to make decisions, since each information source has its advantages and disadvantages. For instance, Social Media sources can provide real-time streaming information, but not all of such information is related to disasters that we are interested in. In fact, there is a high amount of noise in Social Media, which has been elaborated in our previous research study on denial of information [60, 61, 59]. Also, one interesting example of the noise about "landslide" is the 70's rock song "Landslide" by Fleetwood Mac. Twitter filter for the word "landslide" gets more tweets on this song than landslide disasters that involve soil movement. News channels provide reliable and mostly verified information sources. Unfortunately, they normally have high latency that may be up to several days after the occurrence of a disaster.

Besides, disasters like multi-hazards present more significant challenges, since there are no effective physical sensors that would detect multi-hazards directly. Landslide, which can be caused by earthquakes, rainfalls and human activity among other reasons, is an illustrative example of a multi-hazard. After investigating existing approaches using physical and social sensors, we proposed a new landslide detection service – LITMUS [41, 40, 42] and also implemented a prototype system in practice, which is based on a multi-service composition approach to the detection of landslides. More concretely, LITMUS has the following benefits compared with traditional or existing approaches for natural disaster detection:

- It composes information from a variety of sensor networks including both physical sensors (e.g., seismometers for earthquakes and weather satellites for rainfalls) and social sensors (e.g., Twitter and YouTube). Besides providing wider coverage than a system relying on a single source, it improves detection accuracy

and reduces the overall latency.

- It applies state-of-art filters for each social sensor and then adopts geo-tagging to integrate the reported events from all physical and social sensors that refer to the same geo-location. Such integration achieves better landslide detection when compared to an authoritative source. Meanwhile, the geo-location information not only provides the base for the integration, but also enables us to do real-time notification in the future.

- It provides a generic approach to the composition of multiple heterogeneous information services and uses landslide detection as an illustrative example, i.e. it is not tied to disaster detection and can be applied to other application areas involving service composition. Traditional approach to the composition of web services makes strong assumptions about services, which it then uses to select services when composing a new service, such as quality of service [48] or service license compatibility [15]. In practice, the real world services do not satisfy such assumptions. The claim we make is that more information services should provide a more solid result and we demonstrate that it is the case with LITMUS.

The rest of the chapter is organized as follows. Section 3.2 provides an overview of the LITMUS system. We introduce the supported physical and social sources, and describe implementation details of each system component. In Section 3.3, we present an evaluation of landslide detection using real data and compare the results generated by LITMUS with an authoritative source. We summarize related work in Section 3.4 and conclude the chapter in Section 3.5.

## 3.2 System Overview

There are several stages in the LITMUS prototype that are implemented by the corresponding software components – see Figure 7 for an overview of the system

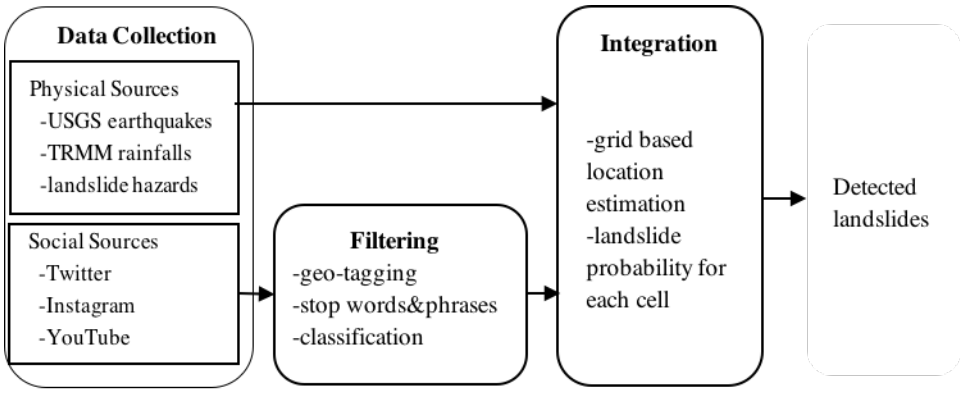


**Figure 7:** Overview of system pipeline

pipeline.

The data collection component downloads the data from multiple social and physical sources using provided API. The data from Social Media requires additional processing as it is usually not geo-tagged and contains a lot of noise. That is why the data from Social Media is geo-tagged followed by the filtering out of irrelevant items using stop words/phrases and classification algorithms. The integration component integrates the data from social and physical sources by performing grid-based location estimation of potential landslide locations followed by the computation of landslide probability to generate a report on detected landslides. This report includes all of the data related to detected landslides, i.e. the physical sensor readings as well as all tweets, images, and videos that were used to detect them.

### 3.2.1 Data Collection Component

**Social Media feeds.** There is a separate data collection process based on the capabilities provided by each data source. Among the currently supported data sources, Twitter has the most advanced API for accessing its data. In particular, it provides a Streaming API, which returns tweets in real-time containing the given keywords. Instead of storing the incoming tweets directly into a data store, LITMUS writes the tweets to a set of intermediate files first. The intermediate layer was introduced for

two reasons. On the one hand we wanted to increase overall robustness, such that even if the data store failed we would still have the original files that we could restore the data from. On the other hand it allows us to easily switch to another data store if needed. The file structure of the intermediate layer is as follows:

```
<source_type>_<event_type>_<year>/<month>/<day>/<hour>/<min>.json
```

Note that when there are multiple incoming items per minute, then they get appended to the same file. The item IDs are used to make sure there are no duplicate records. The rate of incoming items containing landslide keywords is moderate, but we plan to add support for other types of events that would have a much higher rate of incoming items, such as "ebola" for instance. So, a file structure as this makes sure that the data is broken into manageable chunks.

The next step is to upload the incoming items to a data store. We use Redis, because it is an in-memory data store that is widely used and it is open source [50]. We keep the latest 30 days worth of data in the data store to maintain a fixed memory footprint. The new data is periodically uploaded into Redis and obsolete items are removed. The rest of the system works with Redis directly instead of files.

Both YouTube and Instagram provide a pull type of API that LITMUS uses to periodically download items containing landslide keywords. Again, the items from these Social Media get stored into the described file structure and the new items are periodically uploaded into Redis.

**The rainfalls data** is available due to the Tropical Rainfall Measuring Mission (TRMM) [31]. TRMM is a joint space project between NASA and the Japan Aerospace Exploration Agency (JAXA). The mission uses a satellite to collect data about tropical rainfalls. TRMM generates various reports based on its data, including a list of potential landslide areas due to extreme or prolonged rainfall. In particular, it generates reports of potential landslide areas after 1, 3, and 7 days of rainfall. The data is provided in HTML format, which LITMUS periodically downloads, parses

and saves extracted content into data storage for further analysis. TRMM project has been operating since December 1997. However, on July 8, 2014 pressure readings from the fuel tank indicated that the TRMM satellite is near the end of its fuel. The satellite is estimated to be shutdown in February 2016, but JAXA may stop distribution of the radar data prior to that date. As of January 1, 2015 the data is still available.

**The seismic feed** is provided by the United States Geological Survey (USGS) agency [57]. USGS supports multiple feeds of earthquakes with various magnitudes. The data is provided in a convenient GeoJSON format, which is a format for encoding a variety of geographic data structures. LITMUS uses a real-time feed of earthquakes with 2.5 magnitude or higher, which gets updated every minute. USGS includes event id, which is used to avoid duplicate records in the system.

Global Landslide Hazards Distribution is another physical source that LITMUS supports [7]. It provides a 2.5 minute grid of global landslide and snow avalanche hazards based upon the work of the Norwegian Geotechnical Institute (NGI). This source incorporates a range of data including slope, soil, precipitation and temperature among others. The hazard values in this source are ranked from 6 to 10, while the values below are ignored. The reason why this particular source is supported is because the landslides detected by LITMUS to occur in the landslide hazardous areas are more likely to be determined correctly as opposed to the landslides detected to occur in other areas.

### 3.2.2 Filtering Component

**Geo-tagging.** All Social Media supported by LITMUS allow users to disclose their location when they send a tweet, post an image or upload a video. However, based on the evaluation dataset collected in November 2014 very few users actually use this functionality. In particular, less than 0.77% of all tweets are geo-tagged in our

dataset. That is why we analyze the textual descriptions of the items from Social Media to see if they mention geographic terms in them.

A common approach implementing this idea is based on the use of a gazetteer. A gazetteer is a dictionary that maps geographic terms to geographic coordinates. An exact match of a sequence of words is performed against the gazetteer. Since we do not know in advance which particular word or sequence of words is a geographic term, all possible sequences are considered. This approach requires the presence of a local and relatively small gazetteer, since requests to remote or large gazetteers will significantly slow down the system, as the number of sequences of words in a text is very high.

Another weakness of this approach is that gazetteers often have geo terms that are common nouns, so they are used in texts a lot. For example, "Goes" is a city in Netherlands and "Enterprise" is a city in the United States. Most likely both words will be useless geo terms for the purposes of landslide detection and would have to be excluded from consideration by the system. Also, many news sources contain geographic terms in them, such as "Boston Globe" or "Jamaica Observer". A geo-tagging algorithm would have to have a list of news sources in order to ignore such geographic terms automatically.

This is only a small fraction of issues that would have to be addressed in a geo-tagging algorithm based on the use of a gazetteer. Which is why LITMUS implements an alternative approach that employs a natural language processing technique called named entity recognition (NER).

NER implementations locate and classify elements in a text into pre-defined categories, including names of persons, organizations, dates and locations. For geo-tagging purposes LITMUS extracts sequences of words recognized as locations from text. Then it checks the found geo-terms against a local gazetteer. There is an open source project called GeoNames that provides a free gazetteer dump with more than

10 million places[3]. If the geo term is not found there, LITMUS makes a remote call to the Google Geocoding API[4] to obtain corresponding geographic coordinates, i.e. latitude and longitude values.

See Experimental Evaluation section for the results of the geo-tagging analysis performed by LITMUS during the evaluation period.

**Stop words and phrases.** During the process of building the ground truth dataset described below, we noticed that we could almost instantly tell whether a given social item was irrelevant to landslide as a natural disaster or not. There were several common irrelevant topics discussed in Social Media that were easy to spot due to the use of specific words, including "election", "vote", "parliament" and "Fleetwoodmac", e.g.:

- "What does the Republican election landslide mean?: VIRGINIA (WAVY) – What does the Republican landslide in the... http://t.co/2Alrs48SwK"
- "Landslide... and every woman in the Tacoma Dome wept with the beautiful @StevieNicks @fleetwoodmac #fleetwoodmacworldtour"

Another common irrelevant topic is the use of the lyrics from a popular rock song from the 70's to describe a user's mood at the moment, e.g.:

- "Well I've been afraid of changing cause I built my life around you #LandSlide"

In this case instead of a particular stop word, we use excerpts from the lyrics of a popular song as a stop phrase instead.

Stop words and phrases are easy to understand and fast to execute. So, LITMUS attempts to filter out items using stop words and phrases first before applying classification algorithm described next on the remaining items.

[3]http://www.geonames.org/
[4]geocodingAPI

**Classification algorithm.** To decide whether an item from Social Media is relevant or irrelevant to landslide as a natural disaster, we propose the following approach. The textual description of each item is compared against the texts of relevant Wikipedia articles and the texts of irrelevant articles. Then we use the relevance of the article that is most similar to the given item as our decision.

For a list of relevant articles, we use the landslide keywords as Wikipedia concepts, namely landslide, landslip, mudslide, rockfall, and rockslide. These articles are downloaded, parsed and all HTML markup is removed, so that only their content is used for analysis. In addition to these articles, we also use a set of articles describing actual occurrences of landslides, mudslides, and rockslides, including 2014 Pune landslide, 2014 Oso mudslide, and Frank Slide. For a list of irrelevant articles, we use the landslide stop words to download the corresponding Wikipedia articles, namely Landslide victory, Blowout (sports), Election, Landslide (song), and Politics. Similarly, these articles are downloaded, parsed and all HTML markup is removed, so that only their texts are used for analysis.

To compute the distance between social items and these Wikipedia articles we use a formula named after Swiss Professor Paul Jaccard. He compared how similar different regions were based on the following formula:

$$\frac{\textit{Number of species common to the two regions}}{\textit{Total number of species in the two regions}}$$

This formula gives 0 if the sets have no common elements and 1 if they are the same. This is the opposite of what we need as a similarity measure, so we use the following formula instead:

$$\textit{Jaccard distance} = 1 - \frac{Intersection(A, B)}{Union(A, B)},$$

where $A$ and $B$ are the sets that we want to compare.

Each article is converted to a bag of words representation or more precisely a set of words. Each incoming item from Social Media is also converted to a set of words

representation. Now these sets can be used to compute the Jaccard distance between them.

Using this approach we were able to successfully classify items in November 2014. Table 3 below lists the examples of items from Social Media together with the smallest Jaccard distance values and corresponding Wikipedia concepts. See the Experimental Evaluation section for more details on the experiment.

**Table 3:** Examples of classification of items

| Text | Jaccard distance | Wikipedia concept | Decision |
|---|---|---|---|
| Bad weather hampers rescue operations at Sri Lanka's landslide http://t.co/vYYgwRL1S6 #ANN | 0.9916317991631799 | 2014 Pune landslide | 1 |
| Bertam Valley still deadly: After a mudslide claimed four lives and left 100 homeless, the danger is far from ... http://t.co/ZiauH2YVvJ | 0.9913366336633663 | 2014 Oso mudslide | 1 |
| #bjpdrama World's knowledge in 1 hand site: BJP got landslide Will India become a 1 party state like China Russia http://t.co/jGhp1j84az | 0.9847715736040609 | Landslide victory | 0 |

### 3.2.3 Integration Component

Previously the items from social sources have been geo-tagged and classified as either relevant or irrelevant to landslide as a natural disaster. The items from physical sources are already geo-tagged and there is no need to classify them, as they are all considered relevant to landslide as a natural disaster. Now that we have the items' geographic coordinates, namely their latitude and longitude values, we want to integrate the data based on those values. One possible way of doing it is to divide the surface of the planet into cells of a grid. Items from each source are mapped to the cells in this grid based on their latitude/longitude values. Obviously, the size of these cells is important, because it can range from the smallest possible size to the one covering the whole planet. The smaller the cells, the less the chance that related items will be mapped to the same cell. But the bigger the cells, the more events are mapped to the same cell making it virtually impossible to distinguish one event from

another.

Currently we use a 2.5-minute grid both in latitude and longitude, which corresponds to the resolution of the Global Landslide Hazard Distribution described above. This is the maximum resolution of an event supported by the system at the moment.

The total number of cells in our grid is huge as cells are 2.5 minutes in both latitude and longitude, there are 60 minutes per degree, latitude values range from -90 to +90 degrees and longitude values range from -180 to +180 degrees. But the actual number of cells under consideration is much smaller, because LITMUS only analyzes non-empty cells. For example, there are only 1,192 candidate cells during the evaluation month of November 2014 as you can see in the Experimental Evaluation section below.

Next we consider each non-empty cell to decide whether there was a landslide event there. To calculate the probability of a landslide event $w$ in cell $x$, we use the following weighted sum formula as the strategy to integrate data from multiple sources:

$$P(\omega|x) = \sum_i R_j \frac{\sum_j POS_{ij}^x - \sum_j NEG_{ij}^x - \sum_j STOP_{ij}^x}{\sum_i N_i^x},$$

Here, $R_i$ denotes $i$'th sensor's weight or confidence; $POS_{ij}^x$ denotes positively classified items from sensor $i$ in cell $x$, $NEG_{ij}^x$ denotes negatively classified items from sensor $i$ in cell $x$, $STOP_{ij}^x$ denotes the items from sensor $i$ in cell $x$ that have been labeled as irrelevant based on stop words and stop phrases, and $N_i^x$ denotes the total number of items from sensor $i$ in cell $x$.

In our prototype, we use prior $F - measure$ $R$ as the confidence for each sensor, since $F - measure$ provides a balance between precision and recall, namely $F - measure = 2 * \frac{precision * recall}{precision + recall}$. To generate results in the range from 0 to 1, we normalize the values of into a scale between 0 and 1.

Finally, it should be noted that the given formula generates a score between 0 and 1 that can be used to rank all location cells based on the probability of a landslide

occurrence there.

## 3.3 Experimental Evaluation

In this section, we perform an evaluation of LITMUS using real-world data. In particular, we design an experiment to compare the performance of landslide detection by LITMUS versus an authoritative source. We show that LITMUS manages to detect 41 out of 45 events reported by the authoritative source during evaluation period as well as 165 additional locations. We also describe the collection of the ground truth dataset and provide the details of the dataset collected by LITMUS during this period.

### 3.3.1 Evaluation Dataset

We select the month of November 2014 as the evaluation period. Here is an overview of the data collected by LITMUS during this period – see Table 4.

**Table 4:** Overview of evaluation dataset

| Social Media | Raw data | Geo-tagged data |
|---|---|---|
| Twitter | 83909 | 13335 |
| Instagram | 2026 | 460 |
| YouTube | 7186 | 2312 |

For each geo-tagged item, LITMUS also computes its cell based on its latitude and longitude. The total number of cells during the evaluation period is equal to 1,192. Hence, there are 1,192 candidate locations that LITMUS has to mark as either relevant or irrelevant to landslide as a natural disaster.

### 3.3.2 Ground Truth Dataset

In order to collect the ground truth dataset for the month of November, we consider all items that are successfully geo-tagged during this month. For each such geo-tagged item, we compute its cell based on its latitude and longitude values. All cells during

November represent a set of candidate events, which is 1,192 as shown above. Next we group all geo-tagged items from Social Media by their cell values. For each cell we look at each item to see whether it is relevant to landslide as a natural disaster or not. If the item's textual description contains URL, then we look at the URL to confirm the candidate item's relevance to landslides. If the item does not contain a URL, then we try to find confirmation of the described event on the Internet using the textual description as our search query. If another trustworthy source confirms the landslide occurrence in that area then we mark the corresponding cell as relevant. Otherwise we mark it as irrelevant. It should be noted that we consider all events reported by USGS as ground truth as well.

Overall, there are 212 cells that we marked as relevant. The following are a few examples of social activity related to the events in those cells:

- "Landslide on route to Genting Highlands: PETALING JAYA: A landslide occurred at 4.2KM heading towards Genting ... http://t.co/AYfCKy6H2n"
- "Major back up on HWY 403 Toronto bound in Hamilton due to mudslide. ALL lanes closed at 403 between Main & York. http://t.co/QcRJdjydR1"
- "Trains cancelled between Par and Newquay due to landslip http://t.co/IcGsdS3y5r"

### 3.3.3 Comparison of Landslide Detection versus Authoritative Source

In November 2014 USGS posted links to 45 articles related to landslides[5]. LITMUS detects events described in 41 of them, i.e. over 90% of events reported by the authoritative source were detected by our system. In addition to 41 locations described in these articles, LITMUS managed to detect 165 locations unreported by USGS during this period.

Hence, there are only 4 events reported by USGS that were missed by LITMUS

[5]http://landslides.usgs.gov/recent/index.php?year=2014&month=Nov

during this period. Next we provide explanation why LITMUS did not detect the events described in these articles.

Out of these 4 articles, 2 did not report recent natural disasters. In particular, one article suggests that Bilayat grass, also called trap grass, can be used to prevent landslides in the hills of Uttarakhand[6]. The other article describes the reopening of the Haast Pass in New Zealand[7]. It was closed nightly since a major slip last year and it will stay open due to a three-net system that protects the pass against rock fall.

The third article describes a minor event that did not receive much attention in Twitter, Instagram or YouTube. In particular, this article is a link to an image in Wikipedia of a minor rock fall on Angeles Crest Highway in California[8].

Finally, the fourth article is about a route in Costa Rica that remains closed due to recent landslides in that area[9]. There were many tweets on this subject in Spanish, but not much activity in English. LITMUS currently supports English language only, which is why it missed this event. We are already working on adding support for other languages, including Spanish. See Conclusion section for more details.

As we mentioned earlier, LITMUS detected 165 locations unreported by the authoritative source during this period. The reasons why LITMUS manages to detect more landslide events than the authoritative source are twofold. On the one hand we claim that our approach is comprehensive as it is fully automated, so it processes all items from each supported data source as opposed to a manual approach where an expert may miss an event due to a human error or human limits. On the other hand LITMUS integrates multiple sources in its analysis, both physical and social, and we plan to add more sources over time. See Conclusion section for more details.

[6]http://timesofindia.indiatimes.com/city/dehradun/Now-a-grass-that-could-prevent-landslides/articleshow/45196678.cms

[7]http://www.radionz.co.nz/news/regional/258610/pass-reopens-with-rock-fall-protection

[8]http://en.wikipedia.org/wiki/File:Minor_rockfall_on_Angeles_Crest_Highway_2014-11-05.jpg

[9]http://thecostaricanews.com/route-27-remains-closed-due-to-landslides

Overall, LITMUS detected 41 locations reported by USGS and 165 locations more, which is 206 locations out of 212 total ground truth locations, i.e. a landslide detection rate of over 97% during this period.

## 3.4 Related Work

Event analysis using Social Media received a lot of attention from the research community recently. Guy et al. [17] introduced Twitter Earthquake Dispatcher (TED) that gauges public's interest in a particular earthquake using bursts in social activity on Twitter. Sakaki et al. [49] applied machine learning techniques to detect earthquakes by considering each Twitter user as a sensor. Cameron et al. [3] developed platform and client tools to identify relevant Twitter messages that can be used to inform the situation awareness of an emergency incident as it unfolds. Musaev et al. [41, 40, 42] introduced a landslide detection system LITMUS based on integration of multiple social and physical sources. We provide an overview of LITMUS implementation in this work, demonstrate its advantages using a recent evaluation period and describe enhancements made.

Document classification or document categorization is one of the most studied areas in computer science due to its importance. The problem is to assign a document to one or more classes or categories from a predefined set. Sakaki et al. [49] described a real-time earthquake detection system where they classified tweets into relevant and irrelevant categories using a support vector machine based on features such as keywords in a tweet, the number of words, and their context. Musaev et al. [40] improved the overall accuracy of supervised classification of tweets by converting the filtering problem of each item to the filtering problem of the aggregation of items assigned to each event location. Gabrilovich et al. [12, 13] proposed to enhance text categorization with encyclopedia knowledge, such as Wikipedia. Each Wikipedia article represents a concept, and documents are represented in the feature space of words and relevant

Wikipedia concepts. Their Explicit Semantic Analysis (ESA) method explicitly represents the meaning of any text as a weighted vector of Wikipedia-based concepts and identifies the most relevant encyclopedia articles across a diverse collection of datasets. In our work we identify two classes of Wikipedia articles that contain either relevant or irrelevant to landslides articles. Then we use Jaccard distance instead of a weighted vector to find the most similar article to a given social item. Finally we use the article's class as a decision for the social item's relevance to landslides.

Accurate identification of disaster event locations is an important aspect for disaster detection systems. The challenge for Social Media based analysis is that users do not disclose their location when reporting disaster events or that they may use alias or location names in different granularities in messages resulting in inaccurate location information. Cheng et al. [6] proposed and evaluated a probabilistic framework for estimating a Twitter user's city-level location based on the content of tweets, even in the absence of any other geospatial cues. Hecht et al. [22] showed that 34% of users did not provide real location information, and they also demonstrated that a classifier could be used to make predictions about users' locations. Sultanik et al. [55] used an indexed gazetteer for rapid geo-tagging and disambiguation of Social Media texts. Musaev et al. [42] evaluated three geo-tagging algorithms based on the use of gazetteer and named entity recognition approaches. In our work we employ the named entity recognition approach to identify all location entities mentioned in Social Media first. Then we use a public gazetteer to retrieve geographic coordinates for the found locations. If there is no match in the gazetteer, then LITMUS uses the Google Geocoding API to convert locations into geographic coordinates.

## 3.5 Conclusion

In this chapter, we described and evaluated a prototype implementation of a landslide detection system called LITMUS, which combines multiple physical sensors and

Social Media to handle the inherent varied origins and composition of multi-hazards. LITMUS integrates near real-time data from USGS seismic network, NASA TRMM rainfall network, Twitter, YouTube, Instagram as well as a global landslide hazards map. The landslide detection process consists of several stages of Social Media filtering and integration with physical sensor data, with a final ranking of relevance by integrated signal strength. Our results demonstrate that with such approach LITMUS detects 41 out of 45 reported events as well as 165 events that were unreported by the authoritative source during the evaluation period.

As we showed in the Experimental Evaluation section, LITMUS missed four events reported by USGS in November 2014. One of the events did not have much activity in English, but it did receive more attention in Spanish as it occurred in Costa Rica. That is why we are already working on adding support to LITMUS for event detection in other languages, including Spanish and Chinese. The data from Social Media in different languages can be considered as additional data sources, which will increase the coverage of event detection by LITMUS. It should also be noted that different languages have varying amounts of noise depending on the used keywords. For example, we were surprised to find that the overwhelming majority of items in Social Media containing the word "mudslide" in Russian are relevant to mudslide as a natural hazard, which is an interesting fact that we plan to explore.

One of our objectives in this project is to analyze the possibility of predicting landslides in LITMUS. We have been collecting data in LITMUS since August 2013. Our plan is to eventually be able to predict landslide events based on the data from multiple sources, both physical and social. Landslides are an illustrative example of a multi-hazard disaster and we plan to study the possibility of predicting landslides in LITMUS using not only real-time data feeds from multiple sources, but also historical data that we collected.

We also believe that comprehensive and real-time information about landslide

events can be useful not only to government agencies, but also research and journalism communities. That is why we are developing an automated notification system that people and organizations can subscribe to in order to receive real-time information on major landslides. This service will provide all relevant information collected by LITMUS, including tweets, images and videos related to each detected event.

Finally, the prototype landslide detection system LITMUS is live and openly accessible[10], collecting data and displaying detection results in real-time for continued evaluation and improvement of the system.

[10]https://grait-dm.gatech.edu/demo-multi-source-integration/

# CHAPTER IV

# FAST TEXT CLASSIFICATION USING RANDOMIZED EXPLICIT SEMANTIC ANALYSIS

## 4.1 Introduction

Automated document classification or document categorization is an important area in computer science. The problem is to assign a document using its text to one or more classes or categories from a predefined set. This technique is used in various domains, e.g. for detection of disasters like earthquakes [49]. The performance of text classification while maintaining high precision is especially important in case of real-time systems [37].

Our current area of interest is detection of landslides using an integration of multiple sources, including physical sensors and social networks like Twitter, Instagram and YouTube [41, 40, 42]. We use landslide related keywords, e.g. *landslide* and *mudslide*, to download items from social networks as input to our system. The challenge here is that they are polysemous words where one of their meanings is related to our domain and all other meanings are unrelated and introduce noise, including:

- *landslide* as an adjective describing an overwhelming majority of votes or victory: "Japan PM Abe's LDP on track for landslide in December 14 vote - media - World — The Star Online http://t.co/FrTbhnIazw"

- *landslide* as the Fleetwood Mac song "Landslide" from the 1975 album *Fleetwood Mac*: "Well I've been afraid of changing cause I built my life around you #LandSlide"

- *mudslide* as a popular cocktail: "The best dessert I found at Brightspot yesterday, not too sweet! @creamycomfort #baileys #dessert #mudslide #brightspot brightspot"

A state-of-the-art approach in resolving the sense of polysemous words is called Explicit Semantic Analysis (ESA) and it was introduced by Gabrilovich et al. [13]. Their method represents the meaning of a text in a high-dimensional space of concepts derived from Wikipedia, the largest encyclopedia in existence. This approach, however, cannot be used for classification of texts directly due to the high number of dimensions, which is equal to the number of articles in Wikipedia. We propose to use a sample of the Wikipedia dataset instead of the full repository. This allows us to perform classification rapidly without necessarily having to make a large external repository of knowledge tractable first, while leveraging the capabilities of ESA as a superior word sense disambiguator.

This chapter makes the following contributions:

- we introduce a generic approach for fast text classification using randomized explicit semantic analysis based on a random sample of Wikipedia articles (RS-ESA);
- we perform a quantitative evaluation of the proposed RS-ESA approach using real world landslide data collected in December 2014;
- we provide the results of comparison between the RS-ESA approach and the Expert-ESA approach where instead of a random sample of Wikipedia articles we use a set of related articles selected by an expert driven approach;
- we demonstrate the genericity of the proposed approach by successfully applying it to a different problem where factual texts are separated from fictional based on Wikipedia articles and fan fiction stories.

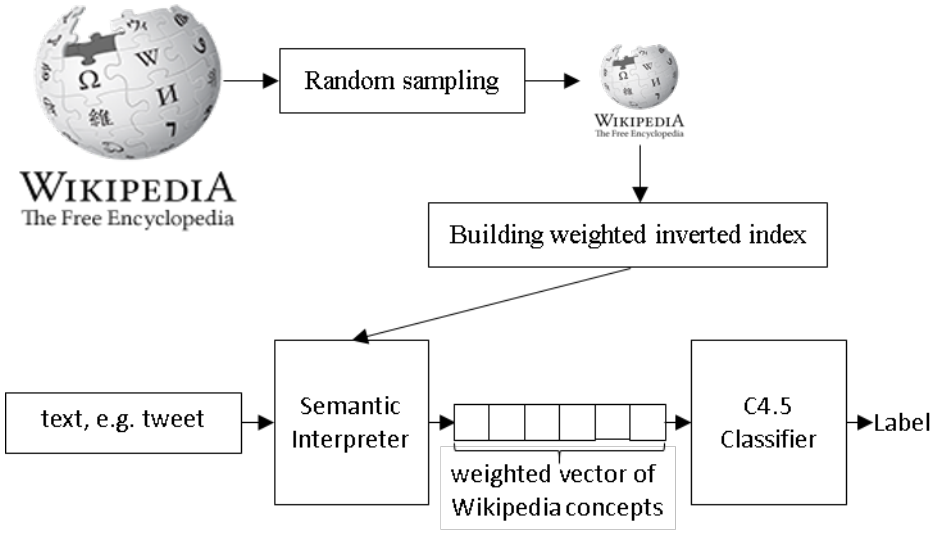


**Figure 8:** RS-ESA overview

The rest of the chapter is organized as follows. We describe the details of the proposed generic classification approach in Section 4.2 followed by the description of the expert based classification approach in Section 4.3. We provide implementation notes in Section 4.4. In Section 4.5 we introduce all datasets that are used for experimental evaluation in Section 4.6. We summarize related work in Section 4.7 and conclude the chapter in Section 4.8.

## 4.2 Randomized Explicit Semantic Analysis (RS-ESA)

As we mention in Section 4.1, Explicit Semantic Analysis (ESA) is the state of the art approach for computing semantic relatedness, but its algorithm is very time-consuming due to the size of the Wikipedia dataset involved. At the moment of writing this publication, there are 4,857,074 articles in the English Wikipedia[1].

To improve the speed of the preprocessing step as well as the throughput of the ESA algorithm, we propose to utilize a sample of the Wikipedia dataset instead of the full dataset similar to the approach used to predict election results. In particular,

[1]http://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia

it is impractical to ask everyone to make a decision and tally the ballots, which would produce 100% accurate results assuming honest answers. Instead, a sample of the population is interviewed in order to get results that reflect the target population as precisely as needed.

The level of precision in this case is affected by two parameters, namely confidence interval and confidence level. A confidence interval is a margin of error. For example, if a confidence interval is 2 and 95% of the sample picked a particular answer then we can be confident that the entire population would have picked that answer between 93% (95−2) and 97% (95+2). The confidence level indicates how sure we want to be. Depending on a problem various values can be utilized, but the most commonly used value is 95%. To determine the sample size for a proportion when sampling without replacement we can use the following equation from statistical inference:

$$n_0 = \frac{Z^2 p(1-p)}{\varepsilon},$$

where $n_0$ is the sample size without considering the finite population correction factor, $Z$ or $Z$-score is a constant that represents the number of standard deviations a given proportion is away from the mean, $p$ is the proportion and $\varepsilon$ is the margin of error.

Applying the finite population correction factor results in the actual sample size $n$ as follows:

$$n = \frac{n_0 N}{n_0 + (N-1)},$$

where $N$ is the population size. Given $Z$-score=1.96 for 95% confidence level, $N$=4,857,074, and $\varepsilon$=0.02, the sample size $n$ should be 2,400.

Our hypothesis is that a sample of the Wikipedia dataset can be used for the ESA method instead of the full dataset to improve its throughput while maintaining high precision. Recall that ESA represents the meaning of any text in terms of Wikipedia-based concepts. Concepts are the titles of Wikipedia articles characterized by the

texts of those articles. In ESA a word is represented as a column vector in the TF-IDF table (table T) of Wikipedia concepts and a document is represented using its interpretation vector, which is a centroid of the column vectors representing its words. An entry $T[i, j]$ in the table of size $N \times M$ corresponds to the TF-IDF value of term $t_i$ in document $d_j$, where $M$ is the number of Wikipedia documents (articles) and $N$ is the number of terms in those documents. See [1] for a more formal description of the ESA method.

For overview of RS-ESA approach - see Figure 8. Note, that we use a decision tree based classifier algorithm C4.5 in our experimental evaluation as we explain in Section 4.6.

## 4.3 Expert Based Explicit Semantic Analysis (Expert-ESA)

As we mention in Section 4.2, we propose to use a random sample of Wikipedia concepts to speed up computations involved in explicit semantic analysis. As an alternative to this approach, we also investigate the use of a subset of Wikipedia repositories selected by an expert driven approach instead of random articles. This approach is thus tied to a particular domain being studied. In our case our domain is landslide detection and we are interested in classification of Social Media data as either relevant or irrelevant to landslide as a natural disaster.

The challenge here is that *landslide* is a polysemous word where one meaning is related to our domain and all other meanings are unrelated and represent noise as described in Section 4.1. That is why we propose to extract a set of articles from Wikipedia that would represent meanings that are relevant to our domain, which is landslide as a natural disaster, and irrelevant meanings. In order to generate a set of articles that describe relevant and irrelevant meanings of our polysemous term, we propose the following approach. For both sets, we start with a list of initial Wikipedia concepts. Each of the Wikipedia articles representing those initial concepts contains

links to other pages inside its text. The articles in these links are used as additional concepts for the corresponding sets. This process can be repeated multiple times to populate our sets of concepts. In this work we follow the links from each of the initial set of articles once and we demonstrate that the total number of concepts obtained this way is sufficient to label items with high precision in Section 4.6 below.

To populate the set of relevant concepts, we can use the set of keywords used to collect landslide data from Social Media as our starting concepts. This set is represented by the following list of Wikipedia concepts: Landslide, Rockfall, Debris Flow, Mudflow, Flash Flood, Earthflow, and Rockslide. Each article representing these concepts contains a list of links to other articles that are also recorded. The total number of concepts extracted using this approach is equal to 550.

To populate the set of irrelevant concepts, we can use the set of Wikipedia concepts that represent the most common reasons for noise in Social Media with respect to landslide as a natural disaster, namely Landslide Victory, Blowout (sports), Landslide (song), Election, List of duo and trio cocktails. The last concept requires some explanation. There is no separate article in Wikipedia on the popular cocktail "Mudslide" as of writing this chapter. However, there is an article listing several cocktails, including Mudslide, so we include that article into a list of irrelevant concepts. Similarly, each article representing these concepts contains a list of links that are also followed. The total number of irrelevant concepts extracted using this approach is equal to 716.

**Table 5:** Overview of dataset for landslide detection

| **Social Media** | **Training Dataset** | **Evaluation Dataset** |
|---|---|---|
| Twitter | 26,953 | 42,268 |
| YouTube | 311 | 466 |
| Instagram | 136 | 204 |

**Table 6:** Overview of dataset of factual and fictional texts

| Data Source | Evaluation Dataset | Class |
|---|---|---|
| Wikipedia articles | 2,400 | Factual |
| FanFiction Twilight Stories | 2,400 | Fictional |

## 4.4 Implementation Details

### 4.4.1 Implementation Notes

To compute a sample size of the Wikipedia dataset for the RS-ESA approach we use the following values: population 4,857,074, confidence interval 2, confidence level 95%. The sample size based on the formulas listed in Section 4.2 is equal to 2,400. In order to select 2,400 random Wikipedia articles we first downloaded a list of all English page titles in main namespace from the Wikipedia dump dated March 4, 2015. Then we randomly selected a title from this list 2,400 times and downloaded a corresponding article using Wikipedia API[2].

Using this sampled dataset we generate table T where columns are titles of the Wikipedia articles, rows are all words present in those articles and $T[i, j]$ elements of the table are TF-IDF values. Note that we apply cosine normalization to each row to disregard differences in document length.

Next for each labeled text in the training and evaluation datasets we compute the centroid of the vectors representing the individual words. The centroid vectors of the training dataset are used to build classifier model, which is then used to predict labels for the centroid vectors of the evaluation dataset.

We perform classification analysis using the Weka software package [18]. Weka is an open source collection of machine learning algorithms and has become the standard tool in the machine learning community.

[2]https://pypi.python.org/pypi/wikipedia/

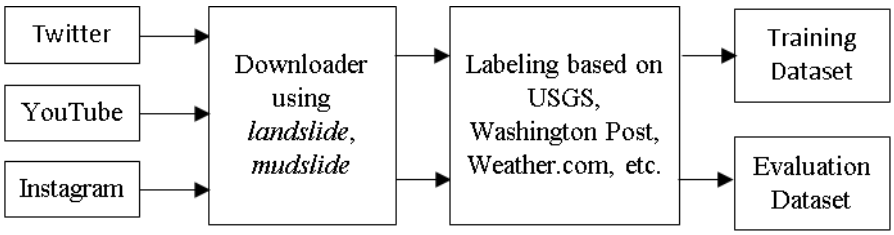


**Figure 9:** Overview of data collection for landslide detection

### 4.4.2 Processing Time

According to the authors of the original ESA approach, parsing of the Wikipedia XML dump on a standard workstation takes about 7 hours on a 2GHz dual core computer, mostly due to the size of the entire Wikipedia corpus at that time. The number of articles in Wikipedia only increased since then. In our approach, the preprocessing step takes less than an hour on a comparable 2.67 GHz computer with 4 cores since we only use a sample of Wikipedia. Although it is a one-time operation, but its processing time still affects the applicability of the approach.

More importantly, the throughput of the original ESA approach is several hundred words per second, whereas RS-ESA's throughput is several thousand words per second, which is an order of magnitude improvement.

## 4.5 Description of Evaluation Datasets

We evaluate the performance of the proposed classification approach using two sets of data. The first dataset is based on the Social Media items collected for landslide detection purposes. The second dataset is based on the Wikipedia articles and FanFiction Twilight stories as sources for classifying texts into factual and fictional categories.

### 4.5.1 Datasets for Landslide Detection Using Social Media

The ground truth dataset for landslide detection includes both training and evaluation datasets - see Table 5. The training dataset contains manually labeled items from

Social Media, namely Twitter, Instagram and YouTube. In particular, it contains values from the *text* field for Twitter, values from the *caption* field for Instagram and values from the *title* and *description* fields for YouTube.

The data for the training dataset was collected during the period from August to December 2013. Labels are either *relevant* or *irrelevant* with respect to landslide as a natural disaster. To prepare a set of relevant items we need a list of confirmed landslides. For this purpose we use expert landslide publications. The USGS agency, in addition to earthquakes, also publishes a monthly list of landslide events collected from external reputable news sources, such as Washington Post, China Daily, Japan Times and Weather.com[3].

To find the Social Media items related to confirmed landslides within each month of the training period, we first filtered the data based on the landslide locations extracted from the confirmed landslides. Then we manually went through each item in the filtered list to make sure they described corresponding landslides by comparing the contents of the items with the corresponding landslide articles. And whenever there were URLs inside those social items, we looked at them also to make sure that they referred to the corresponding landslides. To create a list of unrelated items in the training set, we randomly picked items from each social source and manually went through each item. But this time we had to make sure that the items did not describe landslide events.

The data for the evaluation dataset was collected during the month of December 2014. Labels are again either *relevant* or *irrelevant* with respect to landslide as a natural disaster, but unlike the training dataset all geo-tagged items were labeled. Using the approach described for the training dataset, we identified all items related to the landslides reported by the USGS. Then we analyzed each of the remaining items and followed the URLs to confirm the candidate items' relevance to landslides.

[3]http://landslides.usgs.gov/recent/

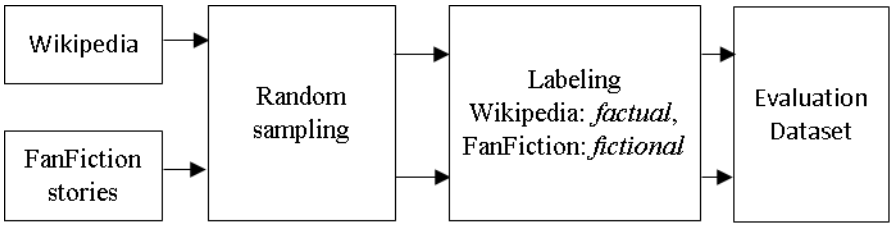


**Figure 10:** Overview of data collection for factual and fictional texts

If the item did not contain a URL, then we tried to find confirmation of the described event on the Internet using its textual description as our search query. If another trustworthy source confirmed the landslide occurrence in the geo-tagged area then we marked the corresponding item as relevant. Otherwise we marked it as irrelevant.

For overview of data collection for landslide detection - see Figure 9. There is a separate downloading process based on the capabilities of each social network. But each downloading process uses the same set of landslide related keywords to retrieve data, including *landslide* and *mudslide*.

### 4.5.2 Dataset for Separation of Factual and Fictional Texts

The ground truth dataset for factual and fictional texts uses two input sources, namely Wikipedia articles and the FanFiction archive of Twilight stories[4]. We consider Wikipedia as a source of factual data and Twilight stories as a source of fictional data.

Our ground truth dataset contains 2,400 Wikipedia articles and 2,400 fan fiction stories. To randomly select 2,400 Wikipedia articles, we again used a list of all English page titles in main space. Then we randomly selected a title from this list 2,400 times. We applied a similar approach to randomly select 2,400 fan fiction stories. First we downloaded 41,851 stories from the FanFiction archive. Note, that we only downloaded the first page of each story to speed up the downloading process. Then we randomly selected an article from this list 2,400 times making sure that the article

[4]https://www.fanfiction.net/book/Twilight/?&srt=1&r=103&p=1

contained at least 100 words.

For overview of data collection for separation of factual and fictional texts - see Figure 10. The labeling process here does not require user input, because we automatically label all Wikipedia articles as *factual* and all FanFiction stories as *fictional*. The experimental evaluation of separation of factual and fictional texts uses 10-fold cross-validation approach, so there is a single evaluation dataset.

## 4.6 Experimental Evaluation

In this section we present an experimental study of the proposed RS-ESA approach and compare it with the Expert-ESA approach. We designed 4 sets of experiments for evaluation purposes. We start by analyzing the effectiveness of RS-ESA for identifying relevance of Social Media data to landslide as a natural disaster using a random sample of Wikipedia repository. To confirm our results we generate a second random sample of Wikipedia repository and perform evaluation of landslide classification again. Next we evaluate Expert-ESA approach and run a third classification analysis of landslide data. Finally, we use RS-ESA approach to perform classification analysis of separating factual texts from fictional.

Note, that we do not include comparison of classification results based on RS-ESA and Expert-ESA approaches versus original ESA, because we were unable to compute a semantic interpreter using the latest Wikipedia XML dump within a reasonable amount of time. However, we intend to add comparisons of both RS-ESA and Expert-ESA versus baseline methods, such as Bag-of-Words approach and others, as part of our future work.

### 4.6.1 Classification of Social Media for Landslide Events

In this and all other experiments we use a decision tree based classifier algorithm C4.5. We choose it, because we want a classifier algorithm to reflect the process of how we built the ground truth dataset for landslide detection described in Section 4.5.

In particular, during the process of manually labeling items from Social Media we noticed that we could almost instantly tell whether a given social item was relevant to landslide as a natural disaster or not. There are several common both relevant and irrelevant topics discussed in Social Media that are easy to spot due to the use of specific words. Each time a particular word was used we could predict with high accuracy the label of the whole text. Hence, we choose a decision tree based algorithm that predicts labels based on the thresholds of the relevance of terms to the concepts represented as features. Note, that Weka's implementation of the C4.5 algorithm is called J48.

For the first experiment we first generated a random sample of 2,400 Wikipedia articles, including:

- Title 1: "Marquetry"
- Title 1,200: "Chemokine receptors"
- Title 2,400: "Shah Kalim Allah Jahanabadi"

Next we generated table T using the words from these articles as rows, titles as columns and the corresponding normalized TF-IDF values as elements. Using this table we computed the centroid vectors for both training and evaluation datasets. Next we used Weka to build a classifier model based on the centroid vectors from the training dataset. Using this model we classified the centroid vectors from the evaluation dataset. For results of classification performance using this approach see row RS-ESA 1 in Table 7. Note, that in spite of a rather low recall of 66% precision is very high at 97%.

To validate high precision results we generated another random sample of 2,400 Wikipedia articles, including:

- Title 1: "980 African Cup of Nations Final"

- Title 1,200: "Macrocneme nigritarsia"
- Title 2,400: "Paleontology in Utah"

For results of classification performance using this sample see row RS-ESA 2 in Table 7. Note, that although precision is a little lower, but it is still quite high at 96%, while recall is higher at 78% and F-score exceeded 86%.

Next we evaluate classification performance using Expert-RSA approach described in Section 4.3. Using the related Wikipedia articles downloaded according to the described method, we generated a new table T using the same approach. Using this table we computed the centroid vectors for both training and evaluation datasets for landslide detection. For results of classification performance using this approach see row Expert-RSA in Table 7. As expected, explicit semantic analysis based on a set of articles selected using an expert driven approach, had a better performance. However, this method requires manual initialization of the starting concepts used to download related articles by an expert user. Also, classification precision achieved using RS-ESA approach is quite high, while not requiring an input from user. It should also be noted that recall using RS-ESA approach is inferior to Expert-ESA, which is why we plan to continue improving RS-ESA performance.

### 4.6.2 Classification of Factual and Fictional Texts

Our final experiment is designed to evaluate the genericity of the RS-ESA approach by classifying data from a different domain. In particular, we choose the problem of classifying texts into factual and fictional categories. For this purpose we use a popular archive of fan fiction, in particular Twilight stories [5]. We consider Wikipedia as a source of factual data and Twilight stories as a source of fictional data.

We reuse the table T generated for RS-ESA 1 experiment. Our evaluation dataset for this experiment is described in Section 4.5. We compute the centroid vectors for

[5] https://www.fanfiction.net/book/Twilight/?&srt=1&r=103&p=1

texts in the evaluation dataset using table T and assign label *factual* to Wikipedia articles and label *fictional* to Twilight stories. We apply 10-fold cross validation using C4.5 classifier and obtain a high value of precision again at 97%.

**Table 7:** Classification of landslide events

| **Approach** | **Precision** | **Recall** | **F-score** |
|---|---|---|---|
| RS-ESA 1 | 97% | 66% | 79% |
| RS-ESA 2 | 96% | 78% | 86% |
| Expert-ESA | 98% | 84% | 91% |

## 4.7 Related Work

Text classification (also known as text categorization, or topic spotting) is used to automatically sort a set of documents into classes (or categories, or topics) from a predefined set [66]. It has attracted a booming interest from researchers in information retrieval and machine learning areas in decades. Recently, several novel classification approaches have been proposed and implemented in text classification. Pu Wang et al. [62] presented semantics-based algorithm for cross-domain text classification using Wikipedia based on co-clustering classification algorithm. Elisabeth Lex et al. [34] described a novel and efficient centroid-based algorithm Class-Feature-Centroid Classifier(CFC) for cross-domain classification of web-logs, also they have discussed the trade-off between complexity and accuracy. Pan et al. [45] proposed a spectral feature alignment (SFA) algorithm to align domain-specific words from different domains into unified clusters, with the help of domain independent words as a bridge. Zhen et al. [67] propose a two-stage algorithm which is based on semi-supervised classification to address the different distribution problem in text classification.

ESA was first introduced by Gabrilovich et al. [13] as an approach to compute the semantic relatedness of terms or short phrases. Since then, lots of researchers have used ESA in many applications successfully. Egozi et al. [10] used ESA for

the estimation of the relevance of documents for a given query and selected high quality features for classification. Potthast et al. [46] and Sorg et al. [53] proposed a cross-lingual extension (CL-ESA) that exploits interlanguage links of Wikipedia articles. Cimiano et al. [8] presented that CL-ESA is superior to other retrieval models which are based on implicit semantics. Also, ESA is used to compute the semantic relatedness of terms. For instance, Mller et al. [39] used ESA as parameters in other retrieval models. In addition, several studies have been conducted to understand or enhance ESA performance [1]. Anderka and Stein revisited ESA and found syntactic parallels to the generalized vector space model (GVSM [65]). They also conducted some initial analysis targeting the impact of the index collection on the performance of ESA. They concluded that the ESA is a general methodology that can be applied on any corpus with concept-level titles or categories. We focus on the Wikipedia use here following several other studies [38, 51]. These studies mostly use Wikipedia corpus to generate concept vectors, and therefore the resulted vector is a vector of Wikipedia concepts given a text document. For example, Scholl et al. (2010) proposed enhancements to ESA (called Extended Explicit Semantic Analysis) that make use of further semantic properties of Wikipedia like article link structure and categorization, thus utilizing the additional semantic information that is included in Wikipedia.

Text mining has been widely used in detection systems for disaster events such as earthquakes and hurricanes. Sakaki et al. [49] proposed an algorithm to monitor tweets and detect earthquake events by considering each Twitter user as a sensor. Cameron et al. [3] developed platform and client tools called Emergency Situation Awareness - Automated Web Text Mining (ESA-AWTM) system by identifying tweets relevant to emergency incidents. Wang et al. [63] proposed a mixture Gaussian model for bursty word extraction in Twitter and then employed a novel time-dependent HDP model for new topic detection. Hua et al. [24] presented STED, a semi-supervised system that helps users to automatically detect and interactively visualize events of a

targeted type from Twitter, such as crimes, civil unrests, and disease outbreaks. Our previous work LITMUS [41, 40, 42] adopts text mining techniques for data analysis on data from multiple information sources such as physical and social information services. To achieve optimized performance of the detection system in terms of precision, we have spent lots of research efforts on improving the text mining techniques in general.

## 4.8 Conclusion

Automated text classification or text categorization is an important problem in computer science. In this chapter we propose a new approach for fast text classification based on randomized explicit semantic analysis (RS-ESA), whose throughput is an order of magnitude faster than the original explicit semantic analysis approach. We demonstrate that our approach using a random sample of Wikipedia articles achieves 96% precision when classifying Social Media landslide data collected in December 2014. We compare the results achieved using RS-ESA approach with explicit semantic analysis approach based on a subset of Wikipedia articles selected by an expert (Expert-ESA) next. Finally, we demonstrate the genericity of the proposed RS-ESA approach by successfully applying it to a different problem where we separate factual texts from fictional based on Wikipedia articles and fan fiction stories, where we achieve 97% precision.

Due to promising results achieved in separating factual texts from fictional using RS-ESA approach based on a limited number of texts, we intend to expand our tests by increasing the size of the evaluation dataset as part of the future work. We plan to add more kinds of sources of factual and fictional texts to confirm our results in diverse domains. We are also interested in evaluating the influence of the sample size on classification performance. Similarly, we are interested in evaluating the influence of the selected concepts used to build ESA table. We plan to run our method multiple

times and report average performance achieved. Finally, we intend to evaluate both Expert-ESA and RS-ESA approaches in other domains.

# CHAPTER V

# REX: RAPID ENSEMBLE CLASSIFICATION SYSTEM FOR LANDSLIDE DETECTION USING SOCIAL MEDIA

## 5.1 Introduction

Social Media platforms have experienced remarkable growth during recent years. For example, there are over 300M active Twitter users monthly that post over 500M tweets per day[1]. These platforms provide active communication channels during mass convergence and emergency events, such as disasters caused by natural hazards [25]. Not only emergency response agencies, but also regular users disseminate situation-sensitive information in safety-critical situations [58]. See Figure 11 for an example of the earliest tweets on Washington State mudslide that occurred on March 22, 2014, when a portion of an unstable hill collapsed, sending mud and debris along an area of approximately 1 square mile. Note, that regular users were one of the first to report about this deadly disaster in Social Media.

We are interested in a particular kind of natural disasters, namely landslides, as they present unique research challenges. Above all, there are no effective physical sensors that would detect landslides directly. In addition, landslide related keywords have multiple meanings that require sophisticated approaches to filter out irrelevant messages. We developed a real-time landslide detection service LITMUS based on our studies and made it openly accessible for continued evaluation and improvement of the system[2]. See Figure 12 for a screenshot of LITMUS in action.

[1] https://about.twitter.com/company

[2] https://grait-dm.gatech.edu/demo-multi-source-integration/

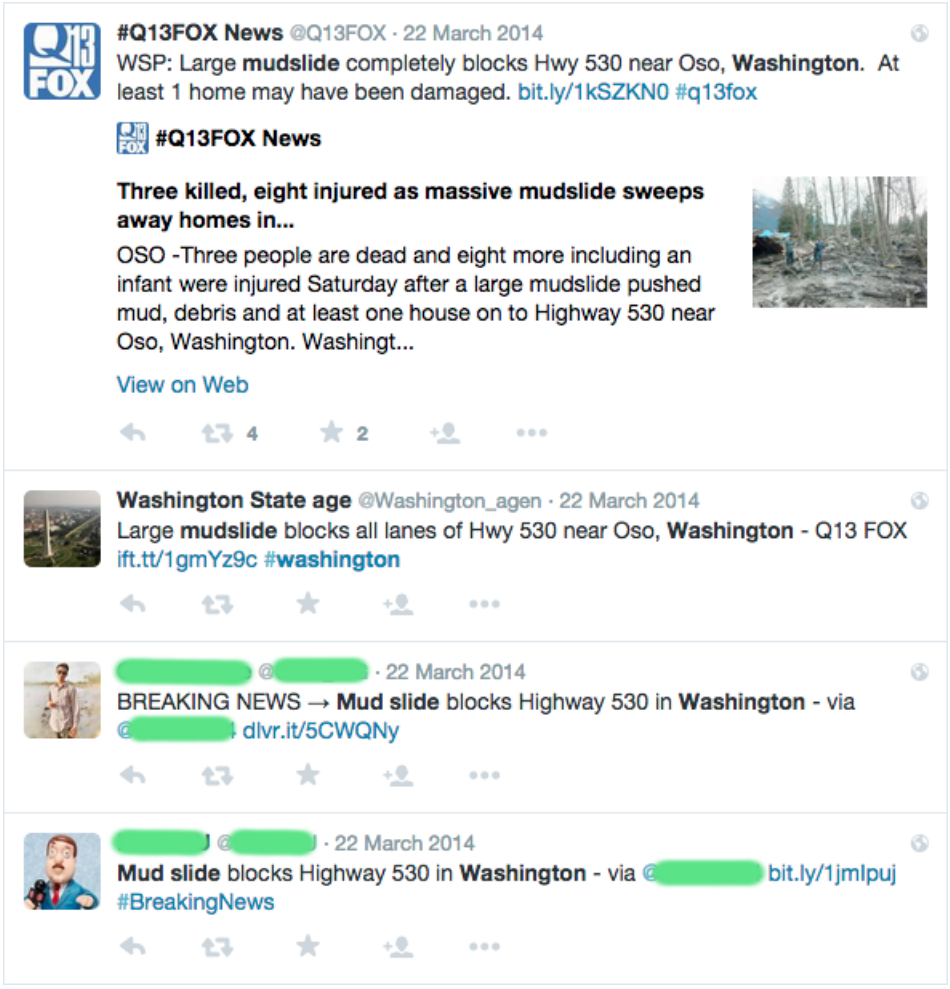


**Figure 11:** Example of disaster related tweets

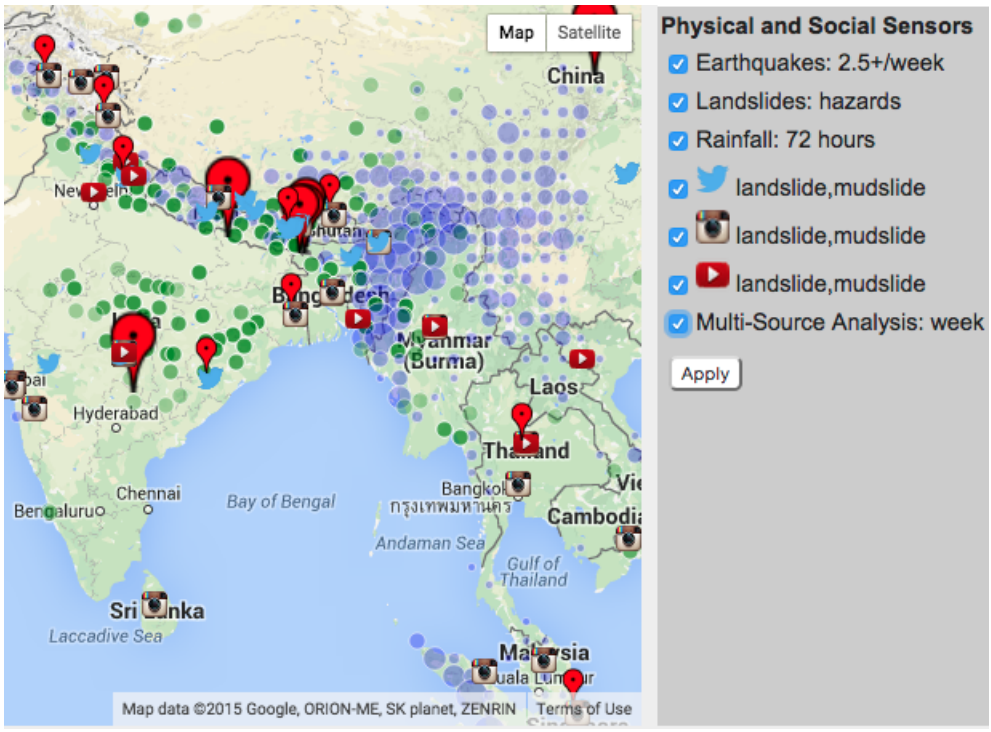


**Figure 12:** LITMUS demonstration

In this chapter, we study the problem of determining the relevance of Social Media texts to landslide as a natural disaster with respect to event detection using machine learning classification. We explore the research challenges next for a better understanding of this problem.

### 5.1.1 Research Challenges

**Lack of Effective Landslide Sensors.** To the best of our knowledge, there are no physical sensors that would detect landslides directly. However, there are information services based on physical sensors that can detect potential causes of landslides. For instance, there is a real-time seismic activity feed provided by the US Geological Survey (USGS) agency[3]. This feed is updated every minute and provides information about earthquakes of various magnitudes. Another potential causes of landslides are heavy rainfalls. Tropical Rainfall Measuring Mission (TRMM)[4] is a project dedicated

[3]http://earthquake.usgs.gov/earthquakes/
[4]http://trmm.gsfc.nasa.gov/

to measuring rainfalls, generating reports based on the satellite data of the areas on the planet that have experienced rainfalls. LITMUS supports both information services as well as Social Media to capture landslide events that attract the public's attention.

**Irrelevant Meanings of Search Keywords.** We use landslide related keywords, including *landslide*, *mudslide*, and *rockslide*, to download raw data from Social Media as input to the system. The challenge is that these keywords are polysemous words, which have multiple irrelevant meanings. The following is a list of frequent examples of irrelevant topics involving the use of these words:

- *landslide* as an adjective describing an overwhelming majority of votes or victory: "New post: Hage Geingob on track to becoming next president of Namibia in election landslide http://t.co/A4QBn2hqvZ"
- *landslide* as a part of the lyrics from the "Bohemian rhapsody" song by Queen: 'Is this a real life? Or is this just fantasy? Caught in a landslide no escape from reality. *Bohemian Rhapsody, Queen".
- *mudslide* as a popular cocktail: "Nothing better on a Thursday than a boat drink. Mudslide topped off with a Patron Cafe floater at Buoy Bar, Point L. . . "
- *rockslide* as a popular dessert: "Rockslide Brownie ice cream should be a thing in Germany"

The most trivial solution for finding irrelevant items in Social Media is by generating a list of stop words with respect to landslide as a natural disaster, such as *election*, *rhapsody*, *cocktail*, and *brownie*, and a list of stop phrases containing excerpts from song lyrics, for example, *"no escape from reality"*. Nonetheless, even after applying this method there are still many unlabeled items and thus more sophisticated approaches are needed to filter out the remaining noise such as the following:

- "Florida State overrated!?! You went too far bruh, they won every game by LANDSLIDES pay respect."

- "President Goodluck Jonathan will run in 2015 and win by a landslide, he saved Nigeria from imminent collapse — Chief Edwin Clark"

**Lack of Geo-tagged Data.** Since a disaster event has a point in time and space, we only consider geo-tagged items in our classification algorithm. Most of the social networks provide support for users to disclose their location when they post a tweet or share a photo. Studies show, however, that less than 0.42% of all tweets actually use this functionality [6]. Furthermore, even geo-tagged tweets often discuss events that occur in other places. Taking into account of these observed phenomena, various approaches have been proposed to infer a location based on the contents of the tweets [64], [33].

We use a Named Entity Recognition (NER) approach to identify all location entities mentioned in Social Media texts. Subsequently, we determine the corresponding geographic coordinates by using the Google Geocoding API. Although the geo-tagging problem is outside the scope of this chapter, it is a very important research challenge in analyzing Social Media data.

### 5.1.2 Our Contributions

To address the challenge of irrelevant meanings of search keywords, we propose REX, a rapid ensemble classification system for accurate classification of Social Media texts in regards to landslide as a natural disaster. REX manages to filter out the remaining noise from social networks based on machine learning classification. Compared with the standard Bag-of-Words and the state-of-the-art distributed word representation approaches to text classification, REX takes advantage of three important considerations: (I) Explicit Semantic Analysis method is proficient at word sense disambiguation, and the reasonable size of index collection is between 1,000 - 10,000

random documents [1]; (II) an ensemble of classifiers performs better than any of its individual members if the classifiers are accurate and diverse [9]; (III) an observation that the predicted relevance of a large event comprising multiple tweets to landslide as a natural disaster is highly accurate.

Based on (I), we select a random sample to reduce the high-dimensionality of Wikipedia knowledge repository used as the index collection by the Explicit Semantic Analysis approach. Specifically, we propose a randomized ESA approach that speeds up text classification while maintaining high performance by utilizing all concepts from the reduced Wikipedia repository as classification features.

Based on (II), we justify the use of ensemble classification by showing that the individual classifiers are accurate and diverse, which is a necessary and sufficient condition. Diversity is achieved by using a combination of two approaches - manipulation of input features and manipulation of training examples, while accuracy of individual classifiers is shown empirically to be high, namely $> 0.8$. We also determine a bound on the number of classifiers needed to predict an aggregate label by majority agreement.

Based on (III), we propose to improve the performance of classification of Social Media texts by assigning the aggregate label of a large event comprising multiple tweets to each individual tweet in it. We run a set of experiments where we define a large event as an event having more than 10, 100 and 1,000 tweets and evaluate the performance of the self-correction approach empirically. In summary, our key contributions are:

**1. Construction of Independent Classifiers.** We propose a new method for constructing independent classifiers, where the input features are built using randomized Explicit Semantic Analysis based on Wikipedia as a knowledge repository and the training examples are selected using the general bootstrapping technique.

**2. Ensemble Classification System with Self Correction.** We increase the

overall classification accuracy by running multiple accurate and diverse classifiers in parallel on each data item using majority agreement for label prediction. We justify a bound on the number of classifiers needed and further improve classification accuracy using a self correction approach based on the accuracy of an aggregate label of an event comprising multiple tweets.

**3. Annotated Social Media Datasets.** We release the annotated datasets used in the experiments containing over 282k labeled items and spanning 1.5 years. The datasets are split into training and evaluation sets and are dedicated to a comprehensive coverage of landslide and related natural disaster events.

**Road Map.** Section 5.2 discusses related work. Section 5.3 formally defines the problem of event detection using Social Media. Section 5.4 presents an overview of our approach REX and its place in the landslide detection service LITMUS. Section 5.5 describes our rapid ensemble classification system and Section 5.6 presents the experimental results using real-world data. Finally, Section 5.7 concludes the chapter.

## 5.2 Related Work

### 5.2.1 Computing Semantic Relatedness using ESA

Explicit Semantic Analysis (ESA) was introduced by [13] as an approach that represented the meaning of texts in a high-dimensional space of Wikipedia-based concepts. Given a text fragment, for example, "Bernanke takes charge", ESA generates the following top concepts that are highly relevant to the input — *Ben Bernanke*, *Federal Reserve*, *Chairman of the Federal Reserve*, *Alan Greenspan* (Bernanke's predecessor), *Monetarism* (an economic theory of money supply and central banking), *inflation* and *deflation* [14].

Since ESA discovery, many researchers have used it successfully in various applications, including [10], [46], and [8]. In addition, several studies have been conducted to

understand or enhance ESA performance, including [1]. Anderka and Stein revisited ESA and conducted some initial analysis targeting the impact of the index collection on the ESA performance. They concluded that ESA is a general methodology that can be applied on any corpus with concept-level titles or categories.

[43] proposed to use a random sample of Wikipedia repository as features for rapid text classification. We continue the study of the ESA method and propose an ensemble classification system based on multiple classifiers, where each classifier's model is generated using a combination of two approaches — manipulation of input features and manipulation of training examples.

### 5.2.2 Computing Semantic Relatedness using Distributed Word Representations

Distributed representation methods have experienced a growth of interest in various domains, including natural language processing. The works by [23], [2], and [47] demonstrated that complex relationships among data can be successfully modeled by learning multiple levels of representation. The state-of-the-art model of distributed representation is Continuous Bag-of-Words Model (CBOW) and Skip-gram model proposed in [36]. These representations can effectively encode dimensions of word similarity and allow vector algebraic operations that support both syntactic: $x_{apple} - x_{apples} \approx x_{car} - x_{cars} \approx x_{family} - x_{families}$ and semantic: $x_{shirt} - x_{clothing} \approx x_{chair} - x_{furniture}$ similarities [52].

Distributed word representations have been successfully used for text classification [29, 32]. However, our proposed approach REX is fundamentally different from the distributed word representation approach. REX uses a sample of Wikipedia concepts as features and represents texts as weighted combinations of the concept vectors corresponding to their words, whereas the distributed word representation learns its features via multiple levels of representation.

We compare an implementation of our approach with text classification based on

distributed word representations and show that REX achieves better performance when classifying Social Media texts for landslide detection using an evaluation period of one year.

### 5.2.3 Event Detection using Social Media

Disaster detection based on social media received abundant attention in the last several years. [17] described Twitter Earthquake Detector (TED) system that infers the level of public interest in a particular earthquake from Twitter activity to decide which earthquakes to disseminate to the public. [49] investigated the real-time nature of Twitter for detection of earthquakes and proposed to apply machine learning classification to clarify whether a tweet is actually referring to an actual earthquake occurrence or not. [3] developed platform and client tools to identify relevant Twitter messages that can be used to inform the situation awareness of an emergency incident as it unfolds. [26] leveraged human-participation through crowdsourcing to perform automatic classification of crisis-related microblog communications in real-time. [42] applied a multi-service composition approach to the detection of landslides.

The focus of this work lies in improving the accuracy of real-time disaster detection by proposing a rapid ensemble classification system and evaluating its performance over a one year period.

## 5.3 Problem Definition

Denote as $E$ the set of all natural disaster events in real life. For a social network $s$, denote as $T_s$ the set of all texts related to disaster events published in the social network $s$, and $\phi_s : T_s \rightarrow E$ the injective function mapping each online text of $s$ to a natural disaster event based on relevance (e.g., location and time). Our Event Detection using Social Media (EDSM) problem is defined as follows.

**Definition 1.** ***Event Detection using Social Media:*** *Given social network set $S$: $\{S_1, S_2, ..., S_n\}$, where $n$ is the number of social networks, the problem of event*

*detection is to find a function f to decide if any potential event picked from any social network $S_i$ corresponds to the same natural disaster event, i.e., $f : T_{S_1} \times T_{S_2} \times ... \times T_{S_n} \rightarrow \{0, 1\}$, such that for any event $e \in T_{S_1} \times T_{S_2} \times ... \times T_{S_n}$, we have*

$$f(e) = \begin{cases} 1, & \textit{if } F(\phi_S(T_S)) >= r; \\ 0, & \textit{otherwise.} \end{cases} \tag{3}$$

Here, function $F$ is our integration strategy, and $r$ is the threshold we set up in the strategy. For example, if $F$ is majority voting strategy which calculates the percentage of majority of votes, then $r$ would be 50%.

It is worth noting that the naive approach to solve this problem is by examining every event set without any attribute filtering in social networks, which results in high computational costs. By using REX, we apply noise filtering and text classification to generate all possible events, and then use integration strategy to optimize the results intelligently through self correction.

## 5.4 Framework Overview

In this chapter, we propose REX, a rapid ensemble classification system. REX itself is part of a landslide detection service LITMUS — see Figure 13 for a high-level overview of the system and note where REX resides in the system's pipeline.

LITMUS downloads data from both physical and social information services and uses its geo-tagging component to assign geographic coordinates to data items based on mentions of places in their textual description. Next, it applies its REX component to label geo-tagged data items as either *relevant* or *irrelevant* to landslide as a natural disaster. Finally, LITMUS integrates the reported events from all physical and social sources that refer to the same geo-location and assigns a score to each location based on its computed relevance to landslides.

We focus on the implementation of the REX component in this chapter. REX is composed of the following three main steps:

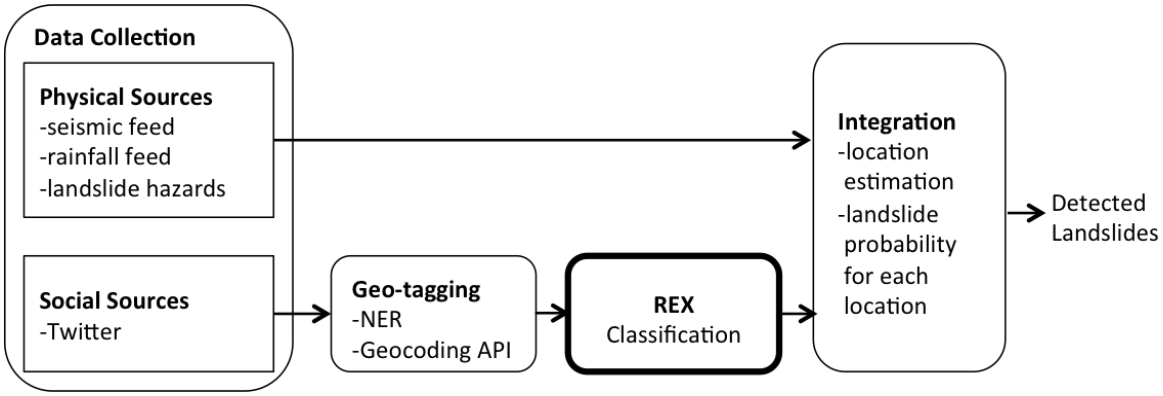


**Figure 13:** Overview of LITMUS pipeline

**Step 1. Construction of individual classifiers.** We propose a randomized Explicit Semantic Analysis approach for building independent classifiers using Wikipedia as a knowledge repository. Details are provided in Section 5.5.1.

**Step 2. Ensemble classification.** We use the Chernoff bounds to determine a bound on the number of classifiers needed to predict an aggregate label by majority agreement. Details are provided in Section 5.5.2.

**Step 3. Self correction.** We take advantage of the observation that the majority label assigned to Social Media texts belonging to a large event is on average correct, in order to improve the performance of ensemble classification. Details are provided in Section 5.5.3.

## 5.5 REX: Rapid Ensemble Classification System

In this Section we describe a new method for constructing independent classifiers that can be used for rapid ensemble classification of Social Media texts, where each classifier is built using randomized Explicit Semantic Analysis. Next we show that these classifiers can be represented as independent Bernoulli random variables. Finally, we use the Chernoff bounds to determine a bound on the number of classifiers needed to predict an aggregate label by majority agreement.

### 5.5.1 Construction of Independent Classifiers

**Formulization of ESA.** As we mention in Section 5.1, Explicit Semantic Analysis (ESA) is a popular method for computing semantic relatedness. The following is a formalization of ESA [53].

Given a text $t$ ESA maps it to a high-dimensional real-valued vector space. This vector space represents external categories, for example Wikipedia repository $W_k = \{a_1, a_2, ..., a_n\}$ in language $L_k$, such that each dimension corresponds to an article $a_i$. This mapping is given by the following function:

$$F_k(t) = \langle v_1, v_2, ..., v_{|W_k|} \rangle,$$

where $|W_k|$ is the number of articles in Wikipedia $W_k$ corresponding to language $L_k$. The value $v_i$ in the ESA vector of $t$ expresses the strength of association between $t$ and the Wikipedia article $a_i$. Based on a function $assoc$ that defines the strength of association between words and Wikipedia articles, the values $v_i$ can be computed as the sum of the association strength of all words of $t = \langle w_1, w_2, ..., w_s \rangle$ to the article $a_i$:

$$v_i = \sum_{w_j \in t} assoc(w_j, a_i)$$

One approach to define such an association strength function $assoc$ is to use a TF-IDF function based on the Bag-of-Words (BOW) model of the Wikipedia articles. The association strength of word $w_j$ to article $a_i$ is then equal to the TF-IDF value of $w_j$ in $a_i$:

$$assoc(w_j, a_i) = TF\text{-}IDF_{a_i}(w_j)$$

Note, that cosine normalization is applied to the association strength function to disregard differences in document length.

**Randomized ESA.** The runtime applicability of the ESA method is challenging due to the size of Wikipedia, which is 4,904,284 articles in its English version as of

this writing and it only keeps growing[5]. The large number of articles makes ESA very hard to use for text classification, because of its high dimensionality. Even the original ESA chapter uses a Bag-of-Words approach enriched with the top 10 concepts instead of applying all Wikipedia concepts for text classification.

[1] shows that the size of a document collection used by ESA method achieves a reasonable trade-off between accuracy and runtime with a number of 1,000 - 10,000 random documents. Therefore, we propose to use a random sample of Wikipedia articles. To determine the sample size for a proportion when sampling with replacement we can use the following equation from statistical inference:

$$n_0 = \frac{Z^2 p(1-p)}{\varepsilon},$$

where $n_0$ is the sample size without considering the finite population correction factor, $Z$ or $Z$-score is a constant that represents the number of standard deviations a given proportion is away from the mean, $p$ is the proportion and $\varepsilon$ is the margin of error.

Applying the finite population correction factor results in the actual sample size $n$ as follows:

$$n = \frac{n_0 N}{n_0 + (N-1)},$$

where $N$ is the population size. Given $Z$-score=1.96 for 95% confidence interval and $\varepsilon$=0.02, $n$ is equal to 2,400, which falls within the range reported by [1].

### 5.5.2 Ensemble Classification

**Justification for Ensemble Classification.** As we show in Section 5.6.4, each individual classifier built using a randomized ESA method has a high accuracy $> 0.8$. We propose to increase an overall accuracy even higher by running multiple classifiers in parallel on each data item, which is an example of an ensemble of classifiers. A necessary and sufficient condition for an ensemble of classifiers to perform better

[5]http://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia

than any of its individual members is if the classifiers are accurate and diverse [9]. An accurate classifier is one whose error rate is better than random guess, which is true in our case as each classifier's accuracy far exceeds 0.5.

Two classifiers are diverse if they make different errors on new data points. Diversity is achieved in REX using a combination of two approaches - manipulation of input features and manipulation of training examples [9]. The randomized ESA approach is an example of input feature manipulation, because Wikipedia articles represent a full set of features and each classifier is built using a random sample of those features.

Manipulation of training examples is achieved by applying the general technique of bootstrapping. Given a training set $X = x_1, x_2, ..., x_n$ with labels $Y = y_1, y_2, ..., y_n$, bootstrapping selects a random sample with replacement of the training set. An example of a bootstrap might be $X = x_2, x_1, x_{10}, x_{10}, ..., x_{821}$ together with the corresponding labels $Y = y_2, y_1, y_{10}, y_{10}, ..., y_{821}$. Note, that there are some duplicates, since a bootstrap resample comes from sampling with replacement. The described approach results in independent training sets and [20] show that the use of independent training sets gives markedly better results than using the same training set for all copies.

See Figure 14 for an overview of how REX classifiers are constructed in parallel. Note, that each classifier uses its own sample of Wikipedia articles for building semantic interpreter and its own sample of the training set.

See Figure 15 for an overview of the classification process performed by REX. It shows that a set of $n$ classifiers is maintained to predict the relevance label of Social Media texts. The majority vote is applied to generate the binary label for each text, where labels are either *relevant* or *irrelevant* to landslide as a natural disaster.

Our approach allows us to classify tweets rapidly compared to the original ESA

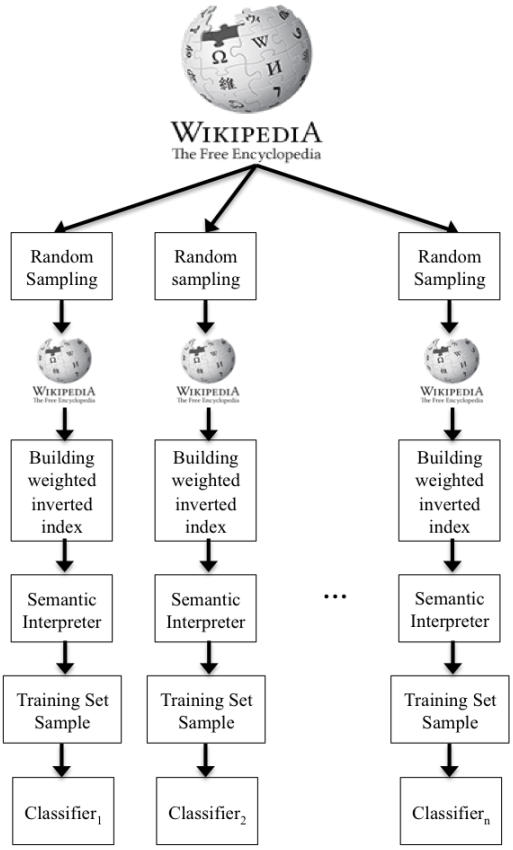


**Figure 14:** Overview of construction of REX classifiers

approach due to a much smaller subset of Wikipedia articles used in REX classification. See the next subsection for a discussion of how to determine the number of classifiers needed for an aggregate decision within a given error rate.

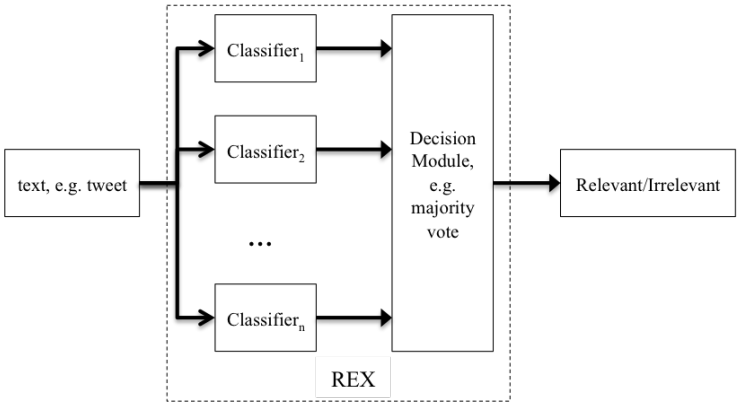


**Figure 15:** Overview of classification process performed by REX

**Bound on the Number of Classifiers.** Recall that a Bernoulli random

variable $X$ is one that takes on the values of 0 and 1 according to

$$P(X = j) = \begin{cases} p, & \text{if } j = 1; \\ q = 1 - p, & \text{if } j = 0. \end{cases}$$

The classifiers in REX can be represented as Bernoully random variables if for each classifier we consider 1 as the correct classification result, which corresponds to either *relevant* or *irrelevant* label and 0 as the incorrect classification result. Furthermore, the classifiers in REX are independent as they are constructed such that each classifier is built using its own set of features and training examples as described in Section 5.5.1.

Let $X_1, X_2, ..., X_n$ be independent Bernoulli random variables, each having probability $p > 1/2$ of being equal to 1. Then the probability of simultaneous occurrence of more than $n/2$ of the events $\{X_k = 1\}$ has an exact value $S$, where

$$S = \sum_{i=\lfloor \frac{n}{2} \rfloor + 1}^{n} \binom{n}{i} p^i (1-p)^{n-i}.$$

The Chernoff bound is the lower bound on S such that

$$S \geq 1 - e^{-2n(p-\frac{1}{2})^2}.$$

Now we can use the Chernoff bounds to bound the success probability of majority agreement for $n$ independent, equally likely events [30]. Suppose we want to ensure that we choose the wrong label with at most a small probability $\varepsilon$. Then

$$\varepsilon \geq e^{-2n(p-\frac{1}{2})^2},$$

$$\ln \varepsilon \geq -2n(p - \frac{1}{2})^2,$$

$$n \geq -\frac{1}{2} \ln \varepsilon \frac{1}{(p-\frac{1}{2})^2},$$

$$n \geq \ln \frac{1}{\sqrt{\varepsilon}} \frac{1}{(p-\frac{1}{2})^2}. \tag{4}$$

Using bound (4) we can compute how many independent trials we need to be confident that we have chosen correctly. For example, given $p = 0.85$ and $\varepsilon = 0.05$, $n$ should be $\geq 13$. In other words, we need at least 13 classifiers running in parallel to predict the label correctly using majority agreement in this case.

### 5.5.3 Self Correction Approach

As we show in Section 5.6.3, the accuracy of REX classification is very high, such that the average F-measure for classification of tweets over a one year evaluation period is ¿ 0.8. We observe that with such accuracy of labeling individual tweets, the aggregate label of an event comprising multiple tweets is also accurate.

We use this observation to improve the performance of REX classification by assigning the aggregate label of a large event comprising multiple tweets to each individual tweet in it. As we show in Section 5.6.4, we define a large event as an event having more than 10, 100 and 1,000 tweets in it and evaluate the performance of the self-correction approach empirically.

We currently use spatiotemporal features of data from physical and social information services for grouping them into events, which can be either relevant or irrelevant. Since an event is a point in time and space, REX only considers geo-tagged items in its classification algorithm. In the future we plan to add topic modeling features to improve location estimation of landslide events.

Note, that the described self correction approach is a generic technique, which can be also utilized by other approaches, including the Bag-of-Words and Word2Vec algorithms, for classification of Social Media items grouped by an event.

## 5.6 Experimental Evaluation

### 5.6.1 Experiment Setup

**Real-World Data.** We conduct a set of experiments using real-world data collected from Twitter. In particular, we use Twitter's Streaming API, which returns tweets

**Table 8:** Overview of datasets

| Type | Relevant | Irrelevant | Total |
|---|---|---|---|
| Training | 13,028 | 13,925 | 26,953 |
| Evaluation | 221,008 | 34,209 | 255,217 |

in real-time using a given set of keywords as its filter[6]. Our keywords include *landslide*, *mudslide*, and *rockslide*. Because we believe that an exact keyword match is performed by the Streaming API, we also include the plural forms of the keywords.

The data contains only geo-tagged items and it is split into training and evaluation sets — see Table 8 for an overview. The training set is collected during the period from August to December 2013 using *landslide* and *mudslide* as search keywords. Note, that an effort is made to have a roughly equal number of relevant and irrelevant training examples. The evaluation set covers the full year of 2014. For this set of experiments we use Social Media texts, namely *text* value of tweets.

Each Social Media text is manually labeled as either *relevant* or *irrelevant*. To mark items as relevant we use two approaches. First we check whether a given text describes a landslide event confirmed by the authoritative source, namely USGS[7]. Each month it compiles a list of landslides that are reported by third party trustworthy sources, including Associated Press, Fox News, Weather Channel, and Reuters. If the landslide event described by a given item is not on this list for a corresponding month then we use a second approach. If the item contains a URL, then we check whether it describes a landslide event and the source is trustworthy. Otherwise, we search for a confirmation of the landslide event online using the described event as a search query.

Irrelevant items are much easier to label as we only need to check whether a given item uses other meanings of the landslide keywords, including overwhelming election victory or an excerpt from the lyrics.

[6]https://dev.twitter.com/streaming/reference/post/statuses/filter
[7]http://landslides.usgs.gov/recent/

We release both annotated datasets as a contribution to the research community[8]. We believe it is the first published dataset that contains annotated data from Twitter covering a 1.5 year period and dedicated to a comprehensive coverage of a particular event type. The datasets are provided in JSON format and contain *item ID*, *cell*, *text*, and *label* fields, where *item ID* is provided by the originating social network, *cell* is the estimated location, *text* is the textual description of the social item and *label* is the manual annotation.

**Experiment Environment.** Our experiments are conducted on a Linux server running Red Hat Enterprise Server 6.5 with Intel(R) Xeon(R) Processor E5640 (12MB Cache, 2.67 GHz, 4 cores), 94 GB Main Memory and 7,200 RPM hard disks.

### 5.6.2 Selection of Classifier Algorithm

**Machine Learning Classifiers.** To find the best algorithm for REX classifiers, we compare various classification algorithms implemented in the Weka software package [18]. Weka is an open source collection of machine learning algorithms that has become the standard tool in the machine learning community. The classifiers evaluated in this experiment include Naïve Bayes, Random Forest, Support Vector Machines, Logistic Regression, and Decision Tree (C4.5). The reason why we select these classifiers is that they are not only popular, but they also represent different categories of classification algorithms. Through those algorithms, we determine which algorithm is the best fit for our ensemble classification system REX.

**Experiment.** For this experiment we generate the vectors using randomized ESA method based on a bootstrap of the training set as described in Section 5.5.1. Then we apply 10-fold cross-validation for each classifier algorithm under consideration, namely Naïve Bayes, Random Forest, Support Vector Machines or SVM (implemented as SMO in Weka), Logistic Regression, and C4.5 (implemented as J48 in Weka) and

[8]https://grait-dm.gatech.edu/resources/

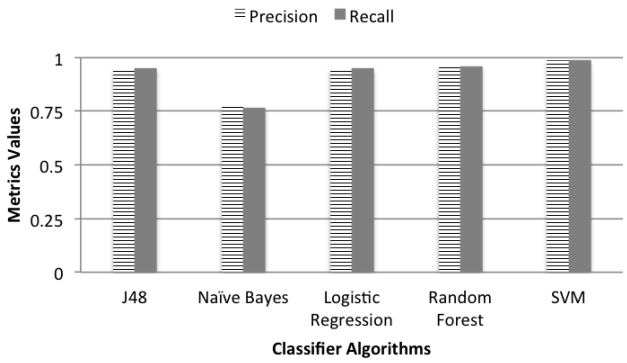


**Figure 16:** Selection of classifier algorithm

report the results in Figure 16.

From Figure 16, we find that SVM shows the best performance out of the five classification algorithms in terms of precision and recall. The result is expected, as SVM is a powerful classifier. Nevertheless, the accuracy of its results is achieved at the expense of execution time. The fastest classifier among the studied algorithms is Naïve Bayes, yet its accuracy is the worst. Based on these results we select SVM to be the base classifier for REX.

### 5.6.3 Comparison of REX vs Baseline Methods

**Baseline Methods.** In the previous experiment described in Section 5.6.2 we show that SVM is the best classification algorithm for landslide detection using Social Media texts in terms of F-measure. Using SVM as the classification algorithm, we now compare REX based ensemble classification against the baseline methods, namely Word2Vec and Bag-of-Words (BOW) models.

As we discuss in Section 5.2.2 the Continuous Bag-of-Words and Skip-gram model is a recent architecture that has been successfully applied in various domains, including text classification. For this experiment we use its Word2Vec implementation[9]. The authors of the implementation have published pre-trained word vectors as part

[9]https://code.google.com/p/word2vec/

of Google News dataset. The model contains 300-dimensional vectors for 3 million words and phrases.

BOW is a common baseline model that represents each document as a bag of words. We select the top 2,400 terms from the training set based on their frequency excluding stop words. We use these terms as features and choose a binary representation based on the presence of each feature in a given text as the weighting scheme.

**Experiment.** For this experiment we evaluate our dataset over the period of one year, 2014 — see Table 8 for an overview. We start with the BOW baseline model. Applying the top 2,400 terms from the training set based on their frequency, we generate the vectors for the training and evaluation sets using binary representation. Next, we build the classification model using SVM as the classifier. We then use the built model to classify each item in the evaluation period.

Afterwards, we continue with the Word2Vec baseline model, based on which we generate vectors for the training set and average vectors for all words in each training example to build the classification model using SVM as the classifier. Similarly we generate vectors for the evaluation set and use the built model to classify each item in the evaluation period.

Finally, we construct SVM classifiers comprising REX according to the approach described in Section 5.5.1. First, we generate 13 Wikipedia samples each containing 2,400 articles randomly selected from Wikipedia. Using these articles as knowledge repositories, we build ESA semantic interpreters. Next we generate training samples using bootstrapping technique and generate corresponding vectors using semantic interpreters to build the SVM classification models. Using the built models, we classify each item in the evaluation period, such that for each item there are 13 predicted labels generated by the corresponding SVM classifiers. We use majority agreement to determine an aggregate label for each item.

Note, that we checked the amount of overlapping articles in 13 Wikipedia samples

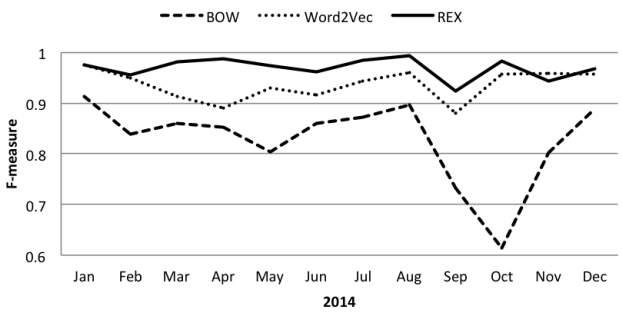


**Figure 17:** Comparison of REX vs baseline algorithms

each containing 2,400 articles. On average there is only 1 overlapping article between each pair of these samples.

See the results of the experiment in Figure 17. REX significantly outperforms BOW based text classification in each month during evaluation period by an average of 0.14 in F-measure. REX also outperforms Word2Vec based classification in F-measure in each month except for November 2014. We plan to conduct further studies to analyze the dip in performance during this month, but overall REX produces better results than Word2Vec by an average of 0.04 in F-measure.

### 5.6.4 REX in Detail

**Ensemble Classification vs Individual Performance.** In this experiment we analyze the performance of the ensemble classification using majority agreement described in Section 5.5.1 and compare it with the performance of the individual classifiers.

The average F-measure performance of classifiers built using randomized Explicit Semantic Analysis is 0.854. As we explain in Section 5.5.2, we need at least 13 classifiers running in parallel to predict the label correctly using majority agreement given the error margin $\varepsilon = 0.05$. We compute the labels of Social Media texts using majority agreement of 13 classifiers and plot the F-measure performance in each month during evaluation period. Next, instead of showing the performance of all

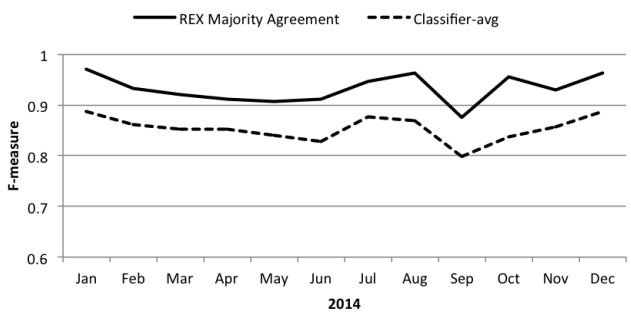


**Figure 18:** Ensemble classification using majority agreement vs average individual performance

individual classifiers we compute their average F-measure performance and plot it in the same diagram.

See the results of this experiment in Figure 18. On average the ensemble classification using majority agreement improved the performance of individual classifiers by 0.078 in F-measure during the evaluation period.

**Influence of Self Correction.** We analyze the influence of the self correction approach described in Section 5.5.3. We use the observation that for large events, both relevant and irrelevant to landslide as a natural disaster, the overall decision by the ensemble classification technique is highly accurate. We define a large event as an event having more than 10, 100 and 1,000 tweets discussing it. We compare the ensemble classification results using majority agreement with the self correction approach and report the results in Figure 19.

Based on this experiment, the self correction approach with large events having 10 or more tweets discussing them demonstrates the best performance. It improves the ensemble classification using majority agreement by an average of 0.042 in F-measure during evaluation period, whereas the self correction approach with large events having 100 and 1,000 items improve the ensemble classification by an average of 0.037 and 0.021 correspondingly.

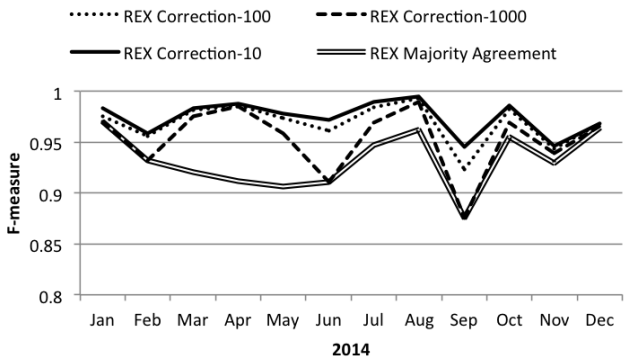


**Figure 19:** Influence of self-correction approach

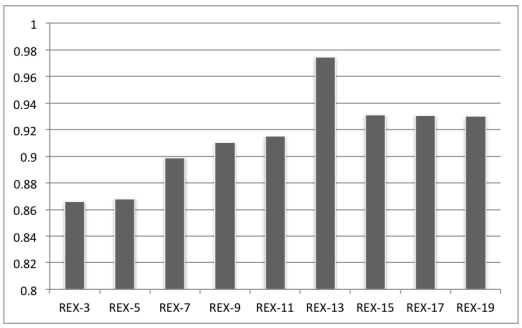


**Figure 20:** Influence of number of classifiers

**Influence of Number of Classifiers.** In this experiment we analyze the performance of REX ensemble classification based on the number of classifiers comprising it. Note, that we use an odd number of classifiers to make sure that we always have a majority agreement winner and there are no undecided cases.

We want to confirm the theoretical bound on the number of classifiers derived in Section 5.5.2. Specifically, the bound is computed to be 13. Hence, we compare the performance of REX ensemble classification with 13 classifiers versus less than 13 classifiers, namely 3, 5, 7, 9, and 11, and greater than 13 classifiers, namely 15, 17, and 19. See the results of the experiment in Figure 20, where we display average F1-scores for each REX-n ensemble of classifiers.

Based on this experiment, the performance of ensemble classification is indeed optimal when the number of classifiers is equal to 13 in our case.

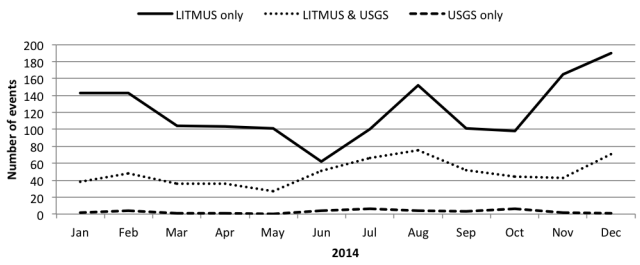


**Figure 21:** Overview of landslide detection results

### 5.6.5 Landslide Detection Results

In this experiment we compare landslide detection results by LITMUS against an authoritative source over a one year period. Specifically, we use a list of landslide events provided by the USGS Landslide Hazards Program[10]. Each month USGS publishes information links for landslide events reported by other reputable sources, including Weather.com, ABC News, China Daily and others. The links also contain the dates when they were posted.

See the landslide detection results in Figure 21. Note, that LITMUS detects the majority of landslides reported by USGS in each month during evaluation period, which is marked as *LITMUS & USGS* in the diagram. In addition, LITMUS detects many more landslides unreported by USGS during the same period — see the *LITMUS only* line in the same diagram. For example, consider December, 2014. During this month LITMUS detects 71 out of 72 landslides reported by USGS as well as 190 landslides unreported by USGS during this period. We plot the example results of landslide detection by LITMUS and USGS during December, 2014 on a Google Map in Figure 22.

Note, that in a few occasions LITMUS misses some landslide events reported by the authoritative source — see the *USGS only* line in the diagram. For example in November, 2014 there was a minor rock fall on Angeles Crest Highway in California[11]

[10]http://landslides.usgs.gov/recent/
[11]http://en.wikipedia.org/wiki/File:Minor_rockfall_on


**Figure 22:** Landslide detection results in December 2014

that was undetected by LITMUS. We find that the main reason for a missed event is the lack of public interest, which can be either due to the fact that it is a minor local event or an event that occurred in a non-English speaking country.

For example, in December, 2014 USGS reported an article about a route in Costa Rica that remained closed due to recent landslides in that area[12]. This event received attention in Social Media in Spanish, but not in English. Currently, LITMUS supports English language only, which is why it missed this event. We are adding support for other languages, including Italian, Portuguese, Russian, Spanish, Chinese and Japanese that we will report about in the next publications.

### 5.6.6 Discussion

The results of the comparison between the proposed system REX and the Word2Vec based method described in Section 5.6.3 may not seem significant at first, but note, that the Word2Vec based text classification using SVM as a classifier algorithm has an average accuracy of 0.94 in F-measure over a one year evaluation period, which

_Angeles_Crest_Highway_2014-11-05.jpg

[12]http://thecostaricanews.com/route-27-remains-closed-due-to-landslides

is a very strong result. That is why although the average accuracy achieved by REX exceeds the accuracy of Word2Vec only by 0.04, the absolute value of 0.98 in F-measure achieved by REX is also a very strong result.

In our experiments we use a trivial approach for determining an aggregate label predicted by an ensemble classification of Social Media texts. As part of our future work we intend to implement a more advanced approach. In particular, instead of using majority agreement, we plan to evaluate a weighted formula, whose weights can be determined based on each classifier's prior performance.

We should also point out that although REX demonstrates better performance than the state-of-the-art approach, but it comes at a computational cost. While the individual classifiers run in parallel, each REX classifier uses vectors with 2,400 features compared to 300 features of the Word2Vec model in our experiment. Thus, SVM classifier implemented in Weka processes 1.3 tweets per second represented as vectors with 2,400 features compared to 12 tweets/sec represented as vectors with 300 features in our experiment environment described in Section 5.6.

Finally, REX is not tied to detection of landslides and we are interested in evaluating REX performance when detecting other natural disasters. Specifically, we are applying REX for detection of harmful algal blooms more commonly known as red tides. The challenge here is that the term "red tide" is also used in social networks to denote irrelevant concepts, e.g. to refer to a team wearing red colors in sports or to describe a propensity for communism in politics.

## 5.7 Conclusion

In this chapter, we filter out noise from Social Media with respect to landslide as a natural disaster. We propose REX, a classification system that constructs independent classifiers, which can be used for ensemble classification of Social Media texts. Each classifier is built using randomized Explicit Semantic Analysis, and is able to

correct its results based on the observation that the majority label assigned to Social Media texts belonging to a large event is highly accurate. We evaluate REX against the standard and state-of-the-art algorithms on the real-world annotated data from Social Media spanning 1.5 years that we release to the public. Experimental results demonstrate that REX achieves high accuracy and outperforms existing algorithms in determining the relevance of Social Media texts to a natural disaster.

# CHAPTER VI

# LOCATION ESTIMATION BASED ON CLUSTERING COMPOSITION

## 6.1 Evaluation of Geo-tagging Algorithms

Most social networks allow users to disclose their location, e.g. when they send a tweet or share a photo, however less than 0.42% of all users actually use this functionality. That is why we implement a geo-tagging component that is responsible for obtaining geographic locations of landslide events. In particular, we retrieve geographic locations based on the mentions of place names that refer to locations of landslides in the item's text.

One of the common approaches implementing this idea is based on the use of a gazetteer, which is a dictionary that maps places to geographic coordinates. An exact match of words in the item's text is performed against such gazetteer. For the list of places we can use the approach introduced by [21], which proposed to use the titles of the geo-tagged Wikipedia articles as its gazetteer. However, different types of geographic locations are supported in the geo-tagged Wikipedia articles. Some of them, like "city" or "country" are more relevant than others, such as "landmark", that often represent irrelevant matches like "houses" or "will".

There are other examples of irrelevant matches, including non-nouns, such as "Says", which was a municipality in Switzerland or "Goes", which is a city in Netherlands. That is why Part-Of-Speech (POS) tagging can be applied to exclude non-noun geo-terms from consideration. There are also geo-terms like "cliff" or "enterprise", whose type is "city", that are irrelevant for our purposes. These words happen to be common nouns that are frequently used in English texts. To mitigate this issue, a

list of most frequent English words can be utilized to exclude such results from the list of geo-terms.

Among the supported social sensors, YouTube in particular contains a lot of items, where in addition to some valuable information related to landslides, they also contain unrelated information. The following is an illustrative example that follows such pattern:

- *"After fatal Flash Flood, Mudslide, More Rain Possible for Colorado and other states youtube original. news bloopers, fox news,onion news,funny news bloopers, news failbreaking news,bbc news news reporter news fails cbs news cnn news world news us news uk news syria today syria war syria 2013 syria new,syria news,damascu,syria damascus, syrian army,syrian,syria execution..."*

It is clear that "Colorado" is a relevant geo-term, whereas "Syria" and "Damascus" are not. To mitigate such issues, we augmented the geo-tagging algorithm as follows: the input text is broken into sentences and for each sentence we find the geo-term that is the closest to the landslide keyword. In this example the landslide keyword is "mudslide" and the closest available geo-term is "Colorado", hence the geo-tagging algorithm correctly outputs "Colorado".

This is just a short list of issues that must be addressed when using geo-tagged Wikipedia articles as a gazetteer. An alternative gazetteer is the Geonames.org database that covers all countries and contains over 10 million places. This implementation also suffers from similar issues as it often extracts irrelevant geographic locations, e.g. "most", "plan" and "cry" that are examples of common nouns.

An alternative approach for extracting geographic locations from social media texts employs a natural language processing (NLP) technique called named entity recognition or NER. Among various entities, NER libraries seek to locate and classify elements in text into pre-defined categories, including names of persons, organizations,

time and location. We are interested in the location entity for the purposes of geo-tagging. LITMUS employs Stanford CoreNLP library, which is a Java suite of NLP tools [11].

Next we provide details of the evaluation of the Wikipedia, Geonames.org and NER based geo-tagging algorithms using real data collected in September 2014.

In total, the Wikipedia based gazetteer approach identified 611 locations in September and out of 55 landslide locations reported by USGS during this period it found 49. Overall, there were 357 landslide locations in September and it found 128 of them.

The Geonames.org based gazetteer approach identified 1568 locations in September and out of 55 landslide locations reported by USGS it found 48. Overall, it was able to identify 153 landslide locations out of 357 during this month.

The NER based approach identified 811 landslide locations and it missed only 3 landslide locations reported by USGS. All of the landslide locations discussed in corresponding social networks in September were successfully identified by this approach. Overall, it found all 357 landslide locations discussed in corresponding social networks.

Overall, the NER based approach produces the least number of irrelevant locations and has the best precision and recall for geo-tagging purposes among the described approaches, which is why LITMUS now implements this approach as part of its geo-tagging component.

## 6.2 Revision of Cell-based Integration

The first step that the integration component performs is it maps the items from each sensor to cells in a grid covering the surface of the Earth. Then it proceeds by considering only non-empty cells. Although this approach is easy to understand and its implementation is fast to compute, it has a few challenges. It is obvious that the size of cells can be either too coarse or too granular for detection purposes, for

example big sized cells will include multiple landslides in them. Another challenge is that it ignores semantics of data items, such that unrelated items may be incorrectly considered as related to the same event and processed together. Let us consider the following items mapped to the same cell and treated in one batch in a cell based approach:

- 3 items on a landslide in Indonesia, including this tweet: "Floods, landslide kill 13 in Indonesia; 2 missing Breaking News MUST SEE. Enjoy the news, subscribe for more!"
- 4 unrelated items from social media mentioning Jakarta, including this Instagram image caption: "Enjoyed this much #creamycomfort #dessert #jakarta #brightspot #baileys #mudslide"

All of these items are mapped to the same cell, because the geo-tagging component returns the same geographic coordinates for both Indonesia and Jakarta. The filtering component classifies these items correctly, but the cell is not deemed a landslide location due to a low integrated landslide score. As this example shows, there are multiple topics connected to the same cell and they should be handled separately. The easiest approach that works in this particular case is to cluster data items based on a geo term within each cell. Such approach correctly detects a landslide in Indonesia. A more advanced approach is to use semantic clustering to group data items with similar content together. This research is described in the following section.

## 6.3 Motivation for Composition of Clustering Algorithms

Not only the data from Social Media contain a lot of noise, but most of the data do not have geo-location either [6]. A common approach for geo-tagging such data is to look for mentions of places in Social Media texts using a gazetteer [55] or a named entity recognition (NER) approach, which generates fewer irrelevant locations [42].

However, even the NER based approach may extract incorrect locations. Consider the following tweet that was posted in December 2014:

- "On the Front Page of Personal Thailand Search for survivors begins after Indonesia landslide kills 18, leaves 90... http://t.co/bcwUzWNqmb"

The NER library incorrectly extracts *Thailand* as the location entity for this tweet, which is an outlier as the location for majority of tweets regarding the disaster event in Indonesia is determined correctly. That is why we propose to cluster Social Media texts based on semantic clustering and to find location outliers for each such cluster.

A further challenge in identifying locations of the detected events is that a single event may comprise multiple locations, which is important to address in order to avoid reporting the same event multiple times. Consider the following tweets mentioning locations affected by mudslide:

- "#LosAngeles News Amid Mudslide Concerns, Glendora Residents Prepare for More Rain: ... http://t.co/VhwIlQ6nCC"

- "Mudslide covers yard of an evacuating resident in Azusa, CA. Taken by @smasunaga: "This is a regulation hoop" http://t.co/xuhVVrHLbx"

Glendora[1] and Azusa[2] are neighboring cities in California that were affected by the same mudslide event, which is why we propose that outlier removal using semantic clustering should be followed by Euclidean clustering, such that locations that are in close proximity to one another are grouped into one cluster. Thus, the second contribution of this chapter is that a composition of clustering algorithms is needed for accurate estimation of locations of the detected events. Based on our knowledge, this is the first work that employs a composition of clustering algorithms to accurately estimate geographic locations based on unstructured texts.

[1]http://cityofglendora.org/about-glendora
[2]http://www.ci.azusa.ca.us/index.aspx?nid=569

## 6.4 Location Estimation Using Semantic Clustering

Majority of items from Social Media do not have geo-location, although each of the supported social sources, namely Twitter, Instagram and YouTube, allow users to disclose their location when they send a tweet, post an image or upload a video. For example, only 0.8% of tweets have geo-location in our evaluation dataset - see Table 9. That is why LITMUS contains a geo-tagging component that attempts to determine the locations of the discussed events by looking for mentions of places in the textual description of the social items. Then it assigns geographic coordinates based on the found geo terms.

In order to find mentions of places in the texts, LITMUS employs an NLP technique called named entity recognition (NER). This technique attempts to recognize various entities in a text, including organizations, persons, dates and locations. We are interested in the location entity for geo-tagging purposes. Once location entities are determined, we can use Google Geocoding API [16] to obtain corresponding geographic coordinates.

LITMUS utilizes Stanford CoreNLP library, which is a Java suite of NLP tools [11], to identify all location entities mentioned in Social Media texts. However, the CoreNLP library occasionally extracts incorrect entities. Consider the following tweet that was posted in December 2014:

- "DTN Mongolia: At least 24 dead in Java landslide: A landslide destroyed a remote village in Java, Indonesia, k... http://t.co/mQUGKYSxWZ"

The NER library incorrectly extracts *Mongolia* as the location entity for this tweet. This is an outlier as for most tweets regarding the disaster event in Indonesia, the library extracts correct geo-terms. That is why we propose to cluster social items based on semantic distance and for each cluster to find such outliers, such that if an overwhelming geo-term exists in a cluster then the location for all social items in the

cluster is set to that geo-term. In this particular example, the overwhelming geo-term in the cluster to which these tweets belong to is *Indonesia*, that is why the location for this tweet is reset by LITMUS accordingly.

## 6.5 Location Estimation Using Euclidean Clustering

In order to estimate locations of landslide events based on data from multiple information services, originally we employed a cell-based approach [41]. The surface of the Earth was represented as a grid of cells and each geo-tagged item was mapped to a cell in this grid based on the item's geographic coordinates.

Obviously, the size of these cells is important. The smaller the cells, the less the chance that related items will be mapped to the same cell. But the bigger the cells, the more events are mapped to the same cell making it virtually impossible to distinguish one event from another. The size we used was a 2.5-minute grid both in latitude and longitude, which corresponds to the resolution of the Global Landslide Hazard Distribution described earlier. That was the maximum resolution of an event supported by the system.

The formulas to compute a cell's row and column based on its latitude (N) and longitude (E) coordinates are as follows:

$$row = (90N)/(2.5'/60') = (90N) * 24 \tag{5}$$

$$column = (180E)/(2.5'/60') = (180E) * 24 \tag{6}$$

For example, Banjarnegara whose geographic coordinates are N = -7.3794368, E = 109.6163185 will be mapped to cell (1983, 6951).

However, a problem with the integration of multiple sources based on cell-based approach is that locations belonging to the same event may be mapped to different cells. This leads to incorrect conclusion that there are multiple events instead of a single one. Consider the following tweets that were posted in December 2014:

- "One village in central Java Banjarnegara Buried landslide - Bubblews http://t.co/iCLRVNNcpG via @GoBubblews"
- "#UPDATE: 12 dead,100 others missing in Jemblung, Indonesia after a landslide was triggered by torrential downpours http://t.co/Npweb5VveG"

The NER library extracts location entity *Banjarnegara* for the first tweet, which is mapped to cell (1983, 6951), and location entity *Jemblung* for the second tweet, which is mapped to cell (1985, 6953). Although the cells are different, but the described event is the same[3]. Jemblung is a village in Banjarnegara regency of Central Java province in Indonesia. These two places are geographically located inside one another even though they are mapped to different cells based on their geographical coordinates.

Hence, we propose to cluster social items based on Euclidean distance instead of solely relying on the cell-based approach to make sure we do not report the same event multiple times. This approach will map tweets that are in close proximity to one another to the same cluster. However, a large number of items from social and physical information services will slow down the execution of a clustering algorithm. For example, our evaluation dataset in December 2014 contains 42k geo-tagged social items. That is why instead of clustering individual items based on their geographic coordinates, we propose to cluster their cells. The total number of candidate cells during the evaluation period is 539, which is significantly less than the number of geo-tagged items. Cells are defined by (row, column) positions that we treat as (X, Y) coordinates for the clustering algorithm based on Euclidean distance.

## 6.6 Evaluation Using Real Data

We select the month of December 2014 as the evaluation period - see Table 9 for an overview of the data collected by LITMUS during this period. Majority of items in

[3]http://news.xinhuanet.com/english/world/2014-12/13/c_133851351.htm

**Table 9:** Overview of evaluation dataset

| Social Media | Raw Data | Data geo-tagged by user | Data geo-tagged by LITMUS |
|---|---|---|---|
| Twitter | 149798 | 1242 (0.8%) | 55054 (36.8%) |
| YouTube | 6533 | 416 (6.4%) | 2749 (42%) |
| Instagram | 4929 | 788 (16%) | 1139 (23.1%) |

each social source do not contain geo-location, which is why we apply the geo-tagging component.

**Table 10:** Evaluation of location estimation

| | Locations based on NER | Locations based on cell-based approach | Locations based on semantic clustering | Locations based on Euclidean clustering |
|---|---|---|---|---|
| Locations | 684 | 539 | 509 | 493 |

The next table contains the results of location estimation based on composition of clustering algorithms - see Table 10. The CoreNLP library detects 684 distinct locations based on Social Media texts from the evaluation dataset. Cell-based approach maps these locations to 539 cells. Semantic clustering removes 5.5% of outlier locations and Euclidean clustering reduces the total number of locations to 493.

Based on the final set of locations generated by the clustering composition approach the actual number of the detected events that were unreported by the authoritative source is equal to 190 instead of 238 landslide locations detected by LITMUS using cell-based approach. This represents a 20% improvement in location estimation due to the proposed clustering composition approach.

# CHAPTER VII

# ANNOTATED DATASET OF LANDSLIDE EVENTS FROM TWITTER

## 7.1 Introduction

Social networking platforms, such as Twitter, have emerged as active communication channels during emergency events, including natural disasters [25]. For example, government agencies disseminate official information to the public via Social Media accounts and even offer digital toolkits to integrate such information into third party tools [5]. Moreover, not only emergency agencies, but also regular users provide situation-sensitive information in safety-critical situations, such as earthquakes [49].

We are interested in a particular kind of natural disasters, namely landslides, as they present unique research challenges. Above all, there are no effective physical sensors that would detect landslides directly. Using data from social networks, such as Twitter, is also challenging due to multiple irrelevant meanings of the word "landslide". It is frequently used as an adjective describing an overwhelming majority of votes or as a reference to the popular 70's rock song of the same name, as opposed to landslide disasters that involve soil movement. In addition, less than 0.4% of tweets are geo-tagged even though Twitter allows users to disclose their location when they post a tweet.

Over the course of our project we collected a dataset of tweets containing landslide related keywords, including *landslide* and *mudslide*. We implemented a geotagging mechanism to extract mentions of geographic terms and retrieved corresponding geographic coordinates. Then, we manually annotated each geotagged tweet based on

its relevance to landslide as a natural disaster. In this chapter we describe our annotated and geotagged dataset of landslide events from Twitter and provide experiments illustrating its usage.

## 7.2 Dataset Overview

Our Twitter dataset covers the full year of 2014. The data is broken into months and stored as separate files. Each file contains JSON formatted strings that provide detailed information about tweets. The information for each tweet includes a set of original attributes returned by Twitter together with a set of custom attributes provided by our system.

See an overview of the dataset in Table 11. The total number of annotated tweets is 255,217. Note, that the overall percentage of geotagged tweets that are labeled as relevant to landslide as a natural disaster is over 86%. Also observe that Social Media users were most active in April and December as they posted the most number of tweets containing keywords "landslide" and "mudslide" during those months.

**Table 11:** Overview of the annotated and geotagged dataset

| Month | Relevant | Irrelevant | Total |
|---|---|---|---|
| 2014-01 | 6,900 | 1,385 | 8,285 |
| 2014-02 | 5,277 | 1,539 | 6,816 |
| 2014-03 | 20,874 | 3,656 | 24,530 |
| 2014-04 | 46,402 | 1,490 | 47,892 |
| 2014-05 | 22,729 | 7,228 | 29,957 |
| 2014-06 | 4,803 | 4,392 | 9,195 |
| 2014-07 | 12,938 | 1,354 | 14,292 |
| 2014-08 | 39,505 | 1,199 | 40,704 |
| 2014-09 | 3,785 | 1,980 | 5,765 |
| 2014-10 | 13,203 | 2,363 | 15,566 |
| 2014-11 | 7,728 | 2,219 | 9,947 |
| 2014-12 | 36,864 | 5,404 | 42,268 |
| 2014 | 221,008 | 34,209 | 255,217 |

## 7.3 Data Collection

We collect data from Twitter using its Streaming API [56]. The Streaming API gives developers low latency access to a stream of tweet data, which is also referred to as status updates. The tweets are pushed in real time to an implementation of a streaming client in JSON format. We use the Streaming API to connect to Twitter's public stream of data and retrieve tweets containing one or more of the given keywords, which include "landslide" and "mudslide". The following are examples of the *text* attribute values of the tweets returned by Twitter:

- Indonesia – Travel News – 2 killed after landslide, triggered by heavy rains, sweeps across Jayapura #Indonesia #Jayapura #travel #safety
- In stand still traffic on 581 South Bound due to an apparent mudslide. And all this time I thought mudslides were only a thing in Indonesia

Twitter returns various attributes in addition to *text*, including:

- *id_str*: unique identifier for this tweet as a string;
- *created_at*: time when this tweet was created in UTC;
- *coordinates*: longitude and latitude values of this tweet's location if it is disclosed, otherwise null;
- *user*: various information about the user, including the UTC datetime when the user's account was created, number of followers and friends, and location.

We provide additional attributes for each tweet as follows:

- *loc*: location entity obtained using Stanford NER library [11];
- *lat*, *lng*: latitude and longitude values retrieved using Google Geocoding API [16] based on the location entity found in the tweet's text;

- *cell*: row and column values of the cell computed using the tweet's latitude and longitude values;
- *label*: manually annotated label based on the tweet's relevance to landslide as a natural disaster.

Attributes *loc*, *lat*, and *lng* are described in Section 7.4, while attribute *label* is described in Section 7.5.

## 7.4 Geotagging Process

Twitter allows users to disclose their location when they post a tweet. However, less than 0.6% of tweets are geo-tagged in our dataset. Furthermore, even if a given tweet is geotagged, the user may be discussing a landslide event that occurred elsewhere, especially in the case of large events. Therefore, we analyze the textual descriptions of the items from Twitter to see if they contain mentions of geographic terms.

In particular, we apply a Named Entity Recognition (NER) approach that locates and classifies elements in text into pre-defined categories, including names of persons, organizations, times and locations. For the purpose of geotagging, we are interested in the location entities mentioned in Social Media texts. Specifically, we use the Stanford NER library [11] to extract mentions of geographic terms in tweets. If there are multiple location entities in a tweet, then we use the geographic term that is the closest to the search keyword.

Given a geographic term, we then determine the corresponding geographic coordinates using the Google Geocoding API [16]. Our next step is to estimate landslide locations based on the retrieved geographic coordinates. The coordinates that are close to one another must be grouped into clusters that represent event locations. We use a cell-based approach for estimating locations [42]. Specifically, the surface of the Earth is represented as a grid of cells. Each geo-tagged item is mapped to a cell in this grid based on the item's geographic coordinates.

**Table 12:** Examples of cell computation

| Tweet | Geo term | Latitude | Longitude | Cell |
|---|---|---|---|---|
| Flood, Storm, Volcano, Mudslide, Indonesia really has it all :( | Indonesia | -0.789275 | 113.921327 | 2141_7054 |
| Seven people missing after flooding and mudslides kill two in Bolivia. #EarthChild | Bolivia | -16.290154 | -63.588653 | 1769_2794 |
| Landslide cut off road for 40,000 in Tamparuli | Tamparuli | 6.14109 | 116.2605431 | 2307_7110 |

It is obvious that the size of the cells plays an important role as it can be either too coarse or too granular for event detection purposes. For example, a single cell covering the whole planet would include all landslide locations, whereas a unit cell would treat each geographic point as a separate event. We choose the size of the cells to be equal to 2.5 minutes in both latitude and longitude, which is roughly equal to 2.875 miles. This size corresponds to the resolution of the Global Landslide Hazard Distribution [7] and represents the maximum resolution supported by our system.

The following formulas are used to compute an item's cell, namely row and column values, given its latitude (N) and longitude (E) coordinates:

$$row = (90N)/(2.5'/60') = (90N) * 24 \tag{7}$$

$$column = (180E)/(2.5'/60') = (180E) * 24 \tag{8}$$

See Table 12 for examples of tweets together with their extracted geographic terms, retrieved coordinates and computed cell values. The cells are represented in the $\langle row \rangle\_\langle column \rangle$ format.

## 7.5 Data Annotation

For annotation purposes we only consider geotagged tweets, because we define a landslide event as a point in time and space. We use two labels, namely *relevant*

and *irrelevant*, in our dataset to indicate whether or not a particular geotagged tweet describes an event that is relevant to landslide as a natural disaster.

Before we started the annotation process, we grouped the tweets by their cell values. This allowed us to significantly speed up the annotation process as the majority of tweets belonging to the same cell discussed the same event, either relevant or irrelevant to landslide as a natural disaster. We also agreed on the definition of relevance to landslide prior to the annotation process. In fact, we labeled any tweet discussing landslides as natural disasters as *relevant* regardless whether they referred to a current, past or future event, or whether they discussed a related topic, e.g. a research project or a grant to assist families impacted by landslide.

Next, during the annotation process, we observed that in majority of cases, it was obvious whether a particular tweet discussed an event that was relevant to landslide or not based on the use of specific words. For example, words like *election*, *vote*, or *fleetwoodmac* would normally indicate an irrelevant topic, whereas words like *victim* or *rockslide* would normally imply a relevant topic. And whenever there were URLs in texts, we viewed them too in order to validate our initial decision based on the contents of the referenced web articles.

See the following examples of tweets discussing topics that are relevant to landslide as a natural disaster:

- Bad weather hampers rescue operations at Sri Lanka's landslide http://t.co/vYYgwRL1S6 #ANN
- Bertam Valley still deadly: After a mudslide claimed four lives and left 100 homeless, the danger is far from ... http://t.co/ZiauH2YVvJ

Similarly, these are the examples of tweets discussing irrelevant topics:

- What does the Republican election landslide mean?: VIRGINIA (WAVY) – What does the Republican landslide in the... http://t.co/2Alrs48SwK

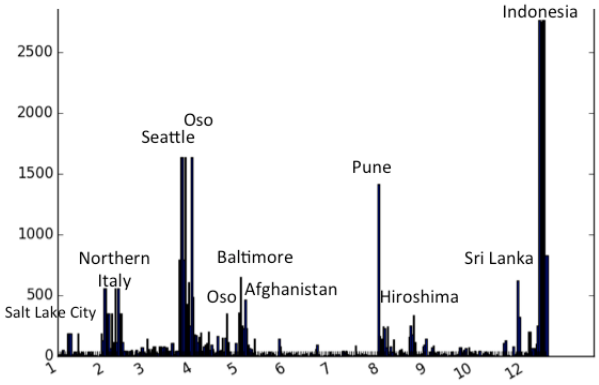


**Figure 23:** Landslide activity in 2014 based on retweets

- Landslide... and every woman in the Tacoma Dome wept with the beautiful @StevieNicks @fleetwoodmac #fleetwoodmacworldtour

The authors annotated each tweet in the geotagged dataset and in case of conflicting labels, we reviewed each case on an individual basis. Discrepancies occurred mainly due to a human error and were easily resolved.

## 7.6 Examples of Usage

### 7.6.1 Visualization of landslide activity based on retweets

In this section we provide a few examples of how our annotated dataset can be used in practice. We first generate a chart of social activity with respect to landslides throughout the year of 2014 — see Figure 23. For each day during that year we record a cell containing the largest number of retweets together with the retweet count. We also add the most common geographic terms associated with those cells for illustration purposes.

Based on the number of retweets, the landslide event that attracted the majority of public interest in 2014 was the mudslide that swept away much of a village in Indonesia[1]. The death toll in this event reached at least 32 people[2].

[1]http://www.foxnews.com/world/2014/12/15/death-toll-from-indonesian-mudslide-rises-to-51-as-rain-halts-search-for-dozens.html?intcmp=latestnews

[2]http://www.reuters.com/article/us-indonesia-landslide-idUSKBN0JR02J20141214

Note, however, that the amount of social activity may not always correspond to the severity of a disaster. For instance, the landslide which buried Afghan village led up to 500 casualties[3]. This event attracted a much smaller number of retweets than the one in Indonesia as seen in the same figure.

### 7.6.2 Evaluation of classification performance

Next we illustrate how this dataset can be used for evaluation of a text classification method. For feature generation purposes we use a distributed word representation approach, which has been successfully used in text classification [29], [32]. Specifically, we apply a state-of-the-art model of distributed representation called Continuous Bag-of-Words and Skip-gram model proposed in [36]. The authors of this model released its implementation called Word2Vec and published 300-dimensional word vectors trained on the Google News dataset using their approach[4].

Here is an overview of how we apply the Word2Vec model to classify tweets. First, we generate vectors for each tweet in our annotated dataset. We divide the generated vectors into training and evaluation sets. Then we build a classification model using a training set. Finally, we use the built model to classify the remaining vectors and evaluate classification performance by comparing the predicted labels versus annotated labels.

We generate vectors using word vectors from the Word2Vec model as follows. For each word in a tweet's text, we retrieve a corresponding vector from the published pre-trained dataset. Then we compute their centroid vector, which is nothing more than the vector obtained by averaging the weights of those vectors [19].

**Selection of classifier algorithm**. In this experiment we select the best classifier algorithm to use for our Word2Vec based model. The classifiers evaluated in

[3]http://www.aljazeera.com/news/asia/2014/05/thousands-reported-dead-afghan-landslide-20145372655196915.html

[4]https://code.google.com/p/word2vec/

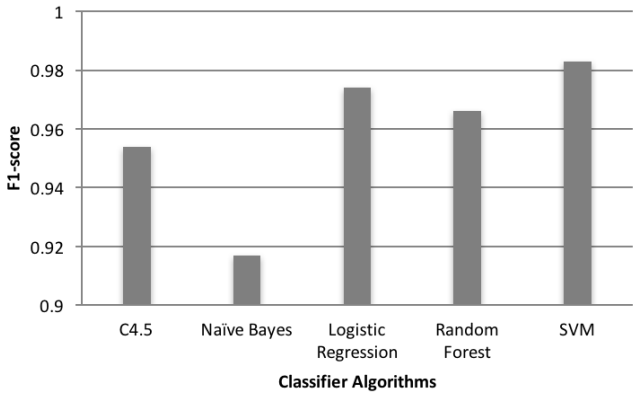


**Figure 24:** Selection of classifier algorithm for Word2Vec

this experiment are Decision Tree (C4.5), Naïve Bayes, Logistic Regression, Random Forest, and Support Vector Machines. We choose these classifiers as they represent different categories of classification algorithms. We use their implementations in the Weka software package to perform our experiments [18].

For this experiment we use the tweets from January, 2014. See the results of the experiment in Figure 24. Here we compute F1-scores for each classifier based on the following formula:

$$F1 = \frac{2 \cdot TP}{2 \cdot TP + FP + FN},$$

where TP stands for true-positive, FP is false-positive, and FN is false-negative labels.

Note, that SVM has the best performance, which is why we select it to be the classifier algorithm for our evaluation.

**Evaluation of Word2Vec classification**. In this experiment we evaluate the classification performance of the Word2Vec based model. We use a standard Bag-of-Words (BOW) model as a baseline algorithm for comparison purposes. BOW represents each document as a bag of words. We select the top 300 words from the training set based on their frequency excluding stop words, so that both approaches use the same number of features. We then use these terms as features and choose a binary representation based on the presence of each feature in a given text as the

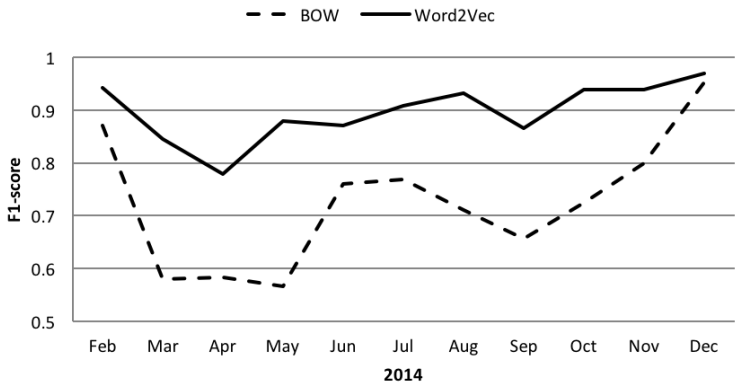


**Figure 25:** Comparison of Word2Vec versus BOW based classification models

weighting scheme. Next we generate vectors for the tweets from the training and evaluation sets, build an SVM model based on the vectors from the training set and use it to classify the vectors from the evaluation period.

Specifically, we select January, 2014 to be the training period and February through December in 2014 to represent the evaluation period. We compute F1-scores for both models and plot the results in Figure 25. Observe that the proposed Word2Vec based classification model consistently outperforms the baseline algorithm.

### 7.6.3 Detection of landslide events

Finally, we demonstrate the detection of landslide events using the described dataset during an evaluation period of November, 2014. In this experiment we plot all of the events detected during this period on a map, so we need to define what we mean by a location. Here the locations are represented by cells in the $\langle row \rangle\_\langle column \rangle$ format, whose row and column values are computed according to equations (7) and (8). We map each tweet to a cell during the evaluation period. Thus, each non-empty cell will have one or more tweets mapped to it.

Each tweet is classified by the Word2Vec classification model described in Section 7.6.2. Hence, each tweet belonging to a cell is assigned a label, which can be either *relevant* or *irrelevant* to landslide as a natural disaster. The simplest strategy

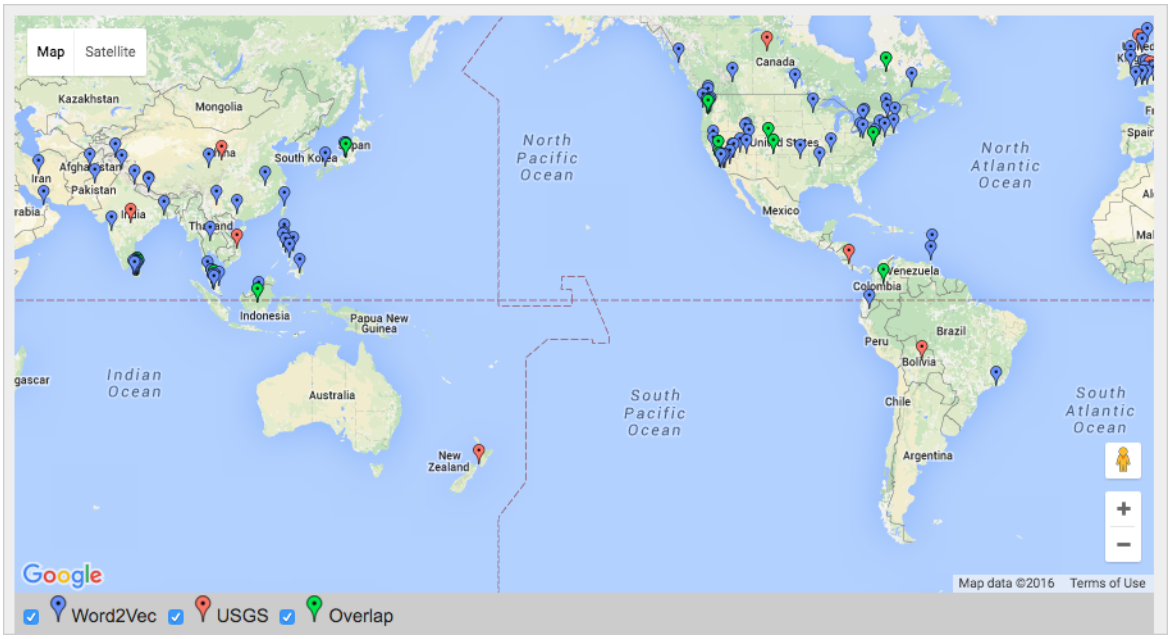


**Figure 26:** Landslide detection results in November, 2014

to decide the relevance of an event represented by a cell is to determine the majority label based on all of the tweets mapped to that cell. In other words, we reward the cell by 1 vote if a given tweet is assigned a *relevant* label and we penalize the cell by 1 vote if a tweet is assigned an *irrelevant* label. If the final score is $\geq 0$ then we believe that the event represented by that cell is relevant to landslide as a natural disaster, otherwise we assume that it is not relevant.

We compare the performance of the proposed method of detecting landslide events versus an authoritative source. In particular, every month the United States Geological Survey (USGS) publishes a list of articles on landslides that occurred during that period[5]. During November, 2014 USGS reported 45 such events[6]. Our Word2Vec based approach managed to detect 36 of them, plus 124 events unreported by USGS. See the results of the experiment in Figure 26.

[5]http://landslides.usgs.gov/recent/
[6]http://landslides.usgs.gov/recent/index.php?year=2014&month=Nov

## 7.7 Conclusion

In this chapter, we describe an annotated dataset of tweets related to landslide events in 2014. The dataset is collected from Twitter using keywords "landslide" and "mudslide". We annotate the tweets with respect to their relevance to landslide as a natural disaster. In addition, the tweets are geotagged based on the presence of mentions of geographic terms in them. This allows for defining landslide events based on their spatiotemporal features. We describe the data collection process and the annotation process, and also explain the process of geotagging the tweets. Finally, we provide several examples of how the released dataset can be used, including visualization of social activity with respect to landslides, evaluation of classification performance and detection of landslide events.

# CHAPTER VIII

# CONCLUSIONS AND FUTURE WORK

In this dissertation, I have provided viable approaches for the integration of physical sensors and social network data to study physical events. As illustration and demonstration, I have built the LITMUS landslide information service that combines data from both physical sensors and social networks in near real-time. More specifically, I have contributed in four parts: integration of multiple sources for landslide detection, filtering out noise from social media, geo-tagging data from social media, and sharing collected data with research community.

In part I, I have introduced a landslide detection service based on a multi-service composition approach (LITMUS) that combines data from both physical and social information services by filtering and then joining the information flow from those services based on their spatiotemporal features. Applying LITMUS to data collected in October 2013, we analyzed and filtered 34.5k tweets, 2.5k video descriptions and 1.6k image captions containing landslide keywords followed by integration with physical sources based on a Bayesian model strategy. It resulted in detection of all 11 landslides reported by USGS and 31 more landslides unreported by USGS.

In part II, I have proposed multiple approaches for determining the relevance of Social Media texts to landslide as a natural disaster. I began with a classification approach based on similarity of texts to two sets of Wikipedia articles describing relevant and irrelevant concepts to landslides. I used the Jaccard distance as a similarity measure and demonstrated that with such approach LITMUS detected 41 out of 45 reported events as well as 165 events that were unreported by the authoritative

source. Next I proposed a new approach for fast text classification using randomized explicit semantic analysis (RS-ESA) to determine the relevance of social media data to landslide as a natural disaster. RS-ESA reduces Wikipedia repository using a random sample approach resulting in a throughput, which is an order of magnitude faster than the original explicit semantic analysis. We demonstrated that our approach achieves 96% precision when classifying social media landslide data collected in December 2014. I have further improved classification performance by proposing a rapid ensemble classification system (REX), which implements two key ideas: 1) a new method for constructing independent classifiers, where each classifier is built using RS-ESA approach; and 2) a self-correction approach which takes advantage of the observation that the majority label assigned to social media texts belonging to a large event is highly accurate. Our experiments using real data from Twitter over a one year period showed that REX classification achieves 0.98 in F-measure, which outperforms the standard Bag-of-Words algorithm by an average of 0.14 and the state-of-the-art Word2Vec algorithm by 0.04.

In part III, I have evaluated three approaches that retrieve geographic locations based on the mentions of place names that refer to locations of landslides in the item's text. We found that the named entity recognition (NER) based approach produces the least number of irrelevant locations and has the best precision and recall for geo-tagging purposes among the evaluated approaches. We improved the quality of the geo-tagging component in LITMUS by proposing a clustering composition approach, where location outliers are removed using clustering based on semantic distance, which is followed by clustering based on Euclidean distance, such that locations that are in close proximity to one another are grouped into the same cluster. Our experiments produced a 20% improvement in location estimation.

In part IV, I have described the released dataset of landslide events from Twitter. The tweets are annotated based on their relevance to landslide as a natural disaster

and geotagged based on the presence of geographic terms in them. The dataset covers the full year of 2014 and it is one of the largest annotated datasets available to date. We provide several illustrations of its possible uses, including visualization of social activity with respect to landslides, evaluation of classification performance and detection of landslide events.

The coverage of physical events detected by LITMUS is currently limited by Social Media data reported in English. However, according to Semiocast[1], only 34% of all tweets in September 2013 were written in English and the second most popular language in Twitter was Japanese. That is why we are adding support for other languages, including Japanese, Chinese, and Hindi. We are also implementing support for additional data sources, such as Sina Weibo as China has no access to Twitter. Sina Weibo is a popular microblogging platform with 222 million monthly active users as of September 2015[2].

Another limitation of our system is that we only analyze textual contents of the data collected from Social Media. That is why our future work will involve the analysis of the image and video contents as they contain valuable information in addition to their textual representation. For example, we observed that there was a common pattern in images depicting landslide and mudslide events. We plan to train machine learning models to identify such images in the future.

We are also interested in improving the precision of event detection. Currently LITMUS assigns the same weight or confidence to each of the data items retrieved from the same source. However, users within the same social network should have different weights depending on their area of expertise, influence as well as the information about their followers and friends among other factors. For example, even if we have no information about a particular user, we can still make some assumption

[1]https://www.statista.com/chart/1726/languages-used-on-twitter/
[2]http://www.chinainternetwatch.com/15740/weibo-q3-2015/

about her based on the users she follows or the users that follow her. If those users reported about landslide events in the past then she is more likely to report relevant information as opposed to irrelevant information about landslides.

Finally, the effect of the use of Wikipedia as the knowledge repository for the reduced explicit semantic analysis in LITMUS should be studied further. Anderka, et al. concluded that the ESA is a general methodology that can be applied on any corpus with concept-level titles or categories [1]. That is why we intend to analyze the use of another popular corpus as the knowledge repository in LITMUS, namely the dataset of Reuters news stories[3].

[3]http://www.daviddlewis.com/resources/testcollections/rcv1/

# REFERENCES


[1] Anderka, M. and Stein, B., "The esa retrieval model revisited," in *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pp. 670–671, ACM, 2009.

[2] Bengio, Y., Lamblin, P., Popovici, D., Larochelle, H., and others, "Greedy layer-wise training of deep networks," *Advances in neural information processing systems*, vol. 19, p. 153, 2007.

[3] Cameron, M. A., Power, R., Robinson, B., and Yin, J., "Emergency situation awareness from twitter for crisis management," in *Proceedings of the 21st international conference companion on World Wide Web*, pp. 695–698, ACM, 2012.

[4] Caragea, C., McNeese, N., Jaiswal, A., Traylor, G., Kim, H. W., Mitra, P., Wu, D., Tapia, A. H., Giles, L., Jansen, B., and others, "Classifying text messages for the haiti earthquake," in *Information Systems for Crisis Response and Management, ISCRAM*, 2011.

[5] Centers for Disease Control and Prevention (CDC) and others, "The health communicator's social media toolkit," *Electronic Media,(July)*, 2011. Accessed on 4/1/2016.

[6] Cheng, Z., Caverlee, J., and Lee, K., "You are where you tweet: a content-based approach to geo-locating twitter users," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, pp. 759–768, ACM, 2010.

[7] CHRR, et al., "Global Landslide Hazard Distribution." `http://sedac.ciesin.columbia.edu/data/set/ndh-landslide-hazard-distribution/`. Accessed on 1/1/2016.

[8] Cimiano, P., Schultz, A., Sizov, S., Sorg, P., and Staab, S., "Explicit versus latent concept models for cross-language information retrieval.," in *IJCAI*, vol. 9, pp. 1513–1518, Citeseer, 2009.

[9] Dietterich, T. G., "Ensemble methods in machine learning," in *Multiple classifier systems*, pp. 1–15, Springer, 2000.

[10] Egozi, O., Gabrilovich, E., and Markovitch, S., "Concept-based feature generation and selection for information retrieval.," in *AAAI*, vol. 8, pp. 1132–1137, 2008.

[11] Finkel, J. R., Grenager, T., and Manning, C., "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pp. 363–370, Association for Computational Linguistics, 2005.

[12] Gabrilovich, E. and Markovitch, S., "Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge," in *AAAI*, vol. 6, pp. 1301–1306, 2006.

[13] Gabrilovich, E. and Markovitch, S., "Computing semantic relatedness using wikipedia-based explicit semantic analysis.," in *IJCAI*, vol. 7, pp. 1606–1611, 2007.

[14] Gabrilovich, E. and Markovitch, S., "Wikipedia-based semantic interpretation for natural language processing," *Journal of Artificial Intelligence Research*, pp. 443–498, 2009.

[15] Gangadharan, G., Weiss, M., DAndrea, V., and Iannella, R., *Service license composition and compatibility analysis*. Springer, 2007.

[16] Google Inc., "The Google Geocoding API." `https://developers.google.com/maps/documentation/geocoding/`. Accessed on 4/1/2016.

[17] Guy, M., Earle, P., Ostrum, C., Gruchalla, K., and Horvath, S., "Integration and dissemination of citizen reported and seismically derived earthquake information via social network technologies," in *Advances in intelligent data analysis IX*, pp. 42–53, Springer, 2010.

[18] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H., "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[19] Han, E.-H. S. and Karypis, G., *Principles of Data Mining and Knowledge Discovery: 4th European Conference, PKDD 2000 Lyon, France, September 13–16, 2000 Proceedings*, ch. Centroid-Based Document Classification: Analysis and Experimental Results, pp. 424–431. Berlin, Heidelberg: Springer Berlin Heidelberg, 2000.

[20] Hansen, L. K. and Salamon, P., "Neural network ensembles," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 10, pp. 993–1001, 1990.

[21] Hecht, B., Hong, L., Suh, B., and Chi, E. H., "Tweets from justin bieber's heart: the dynamics of the location field in user profiles," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 237–246, ACM, 2011.

[22] Hecht, B. J. and Gergle, D., "On the localness of user-generated content," in *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pp. 229–232, ACM, 2010.

[23] Hinton, G. E., Osindero, S., and Teh, Y.-W., "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[24] Hua, T., Chen, F., Zhao, L., Lu, C.-T., and Ramakrishnan, N., "Sted: semi-supervised targeted-interest event detectionin in twitter," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1466–1469, ACM, 2013.

[25] Imran, M., Castillo, C., Diaz, F., and Vieweg, S., "Processing social media messages in mass emergency: a survey," *ACM Computing Surveys (CSUR)*, vol. 47, no. 4, p. 67, 2015.

[26] Imran, M., Castillo, C., Lucas, J., Meier, P., and Vieweg, S., "Aidr: Artificial intelligence for disaster response," in *Proceedings of the companion publication of the 23rd international conference on World wide web companion*, pp. 159–162, International World Wide Web Conferences Steering Committee, 2014.

[27] Imran, M., Elbassuoni, S. M., Castillo, C., Diaz, F., and Meier, P., "Extracting information nuggets from disaster-related messages in social media," *Proc. of ISCRAM, Baden-Baden, Germany*, 2013.

[28] Kanamori, H., "Real-time seismology and earthquake damage mitigation," *Annu. Rev. Earth Planet. Sci.*, vol. 33, pp. 195–214, 2005.

[29] Kim, Y., "Convolutional neural networks for sentence classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1746–1751, 2014.

[30] Kononenko, I. and Kukar, M., *Machine learning and data mining: introduction to principles and algorithms*. Horwood Publishing, 2007.

[31] Kummerow, C., Barnes, W., Kozu, T., Shiue, J., and Simpson, J., "The tropical rainfall measuring mission (trmm) sensor package," *Journal of atmospheric and oceanic technology*, vol. 15, no. 3, pp. 809–817, 1998.

[32] Le, Q. V. and Mikolov, T., "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, pp. 1188–1196, 2014.

[33] Lee, R. and Sumiya, K., "Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection," in *Proceedings of the 2nd ACM SIGSPATIAL international workshop on location based social networks*, pp. 1–10, ACM, 2010.

[34] Lex, E., Seifert, C., Granitzer, M., and Juffinger, A., "Efficient cross-domain classification of weblogs," *International Journal of Intelligent Computing Research*, vol. 1, no. 1, pp. 36–45, 2010.

[35] Mathioudakis, M. and Koudas, N., "Twittermonitor: trend detection over the twitter stream," in *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, pp. 1155–1158, ACM, 2010.

[36] Mikolov, T., Yih, W.-t., and Zweig, G., "Linguistic regularities in continuous space word representations.," in *HLT-NAACL*, pp. 746–751, 2013.

[37] Miltsakaki, E. and Troutt, A., "Real-time web text classification and analysis of reading difficulty," in *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, pp. 89–97, Association for Computational Linguistics, 2008.

[38] Minier, Z., Bodo, Z., and Csato, L., "Wikipedia-based kernels for text categorization," in *Symbolic and Numeric Algorithms for Scientific Computing, 2007. SYNASC. International Symposium on*, pp. 157–164, IEEE, 2007.

[39] Müller, C. and Gurevych, I., "Semantically enhanced term frequency," in *Advances in Information Retrieval*, pp. 598–601, Springer, 2010.

[40] Musaev, A., Wang, D., Cho, C. A., and Pu, C., "Landslide detection service based on composition of physical and social information services," in *Web Services (ICWS), 2014 IEEE International Conference on*, pp. 97–104, IEEE, 2014.

[41] Musaev, A., Wang, D., and Pu, C., "Litmus: Landslide detection by integrating multiple sources," in *11th International Conference Information Systems for Crisis Response and Management (ISCRAM)*, 2014.

[42] Musaev, A., Wang, D., and Pu, C., "Litmus: A multi-service composition system for landslide detection," *Services Computing, IEEE Transactions on*, vol. 8, no. 5, pp. 715–726, 2015.

[43] Musaev, A., Wang, D., Shridhar, S., and Pu, C., "Fast text classification using randomized explicit semantic analysis," in *Information Reuse and Integration (IRI), 2015 IEEE International Conference on*, pp. 364–371, IEEE, 2015.

[44] Palen, L., Anderson, K. M., Mark, G., Martin, J., Sicker, D., Palmer, M., and Grunwald, D., "A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters," in *Proceedings of the 2010 ACM-BCS visions of computer science conference*, p. 8, British Computer Society, 2010.

[45] Pan, S. J., Ni, X., Sun, J.-T., Yang, Q., and Chen, Z., "Cross-domain sentiment classification via spectral feature alignment," in *Proceedings of the 19th international conference on World wide web*, pp. 751–760, ACM, 2010.

[46] Potthast, M., Stein, B., and Anderka, M., "A wikipedia-based multilingual retrieval model," in *Advances in Information Retrieval*, pp. 522–530, Springer, 2008.

[47] Poultney, C., Chopra, S., Cun, Y. L., and others, "Efficient learning of sparse representations with an energy-based model," in *Advances in neural information processing systems*, pp. 1137–1144, 2006.

[48] Ran, S., "A model for web services discovery with qos," *ACM Sigecom exchanges*, vol. 4, no. 1, pp. 1–10, 2003.

[49] Sakaki, T., Okazaki, M., and Matsuo, Y., "Earthquake shakes twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web*, pp. 851–860, ACM, 2010.

[50] Sanfilippo, S. and Noordhuis, P., "Redis," 2009.

[51] Scholl, P., Böhnstedt, D., García, R. D., Rensing, C., and Steinmetz, R., "Extended explicit semantic analysis for calculating semantic relatedness of web resources," in *Sustaining TEL: From Innovation to Learning and Practice*, pp. 324–339, Springer, 2010.

[52] Socher, R. and Manning, C. D., "Deep learning for NLP (without magic)," in *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pp. 1–3, 2013.

[53] Sorg, P. and Cimiano, P., "Cross-lingual information retrieval with explicit semantic analysis," in *Working Notes for the CLEF 2008 Workshop*, 2008.

[54] Starbird, K., Muzny, G., and Palen, L., "Learning from the crowd: collaborative filtering techniques for identifying on-the-ground twitterers during mass disruptions," in *Proceedings of 9th International Conference on Information Systems for Crisis Response and Management, ISCRAM*, 2012.

[55] Sultanik, E. A. and Fink, C., "Rapid geotagging and disambiguation of social media text via an indexed gazetteer," *Proceedings of ISCRAM*, vol. 12, pp. 1–10, 2012.

[56] Twitter, Inc., "Streaming API." `https://dev.twitter.com/streaming/overview`. Accessed on 1/1/2016.

[57] United States Geological Survey, "Earthquakes Hazards Program." `http://earthquake.usgs.gov`. Accessed on 1/1/2015.

[58] Vieweg, S., Hughes, A. L., Starbird, K., and Palen, L., "Microblogging during two natural hazards events: what twitter may contribute to situational awareness," in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1079–1088, ACM, 2010.

[59] Wang, D., "Analysis and detection of low quality information in social networks," in *Data Engineering Workshops (ICDEW), 2014 IEEE 30th International Conference on*, pp. 350–354, IEEE, 2014.

[60] Wang, D., Irani, D., and Pu, C., "A social-spam detection framework," in *Proceedings of the 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, pp. 46–54, ACM, 2011.

[61] Wang, D., Irani, D., and Pu, C., "A study on evolution of email spam over fifteen years," in *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on*, pp. 1–10, IEEE, 2013.

[62] Wang, P., Domeniconi, C., and Hu, J., "Cross-domain text classification using wikipedia.," *IEEE Intelligent Informatics Bulletin*, vol. 9, no. 1, pp. 5–17, 2008.

[63] Wang, X., Zhu, F., Jiang, J., and Li, S., "Real time event detection in twitter," in *Web-Age Information Management*, pp. 502–513, Springer, 2013.

[64] Watanabe, K., Ochi, M., Okabe, M., and Onai, R., "Jasmine: a real-time local-event detection system based on geolocation information propagated to microblogs," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 2541–2544, ACM, 2011.

[65] Wong, S. M., Ziarko, W., and Wong, P. C., "Generalized vector spaces model in information retrieval," in *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 18–25, ACM, 1985.

[66] Zanasi, A., *Text mining and its applications to intelligence, CRM and knowledge management*. Wit Press Southampton Boston, 2005.

[67] Zhen, Y. and Li, C., "Cross-domain knowledge transfer using semi-supervised classification," in *AI 2008: Advances in Artificial Intelligence*, pp. 362–371, Springer, 2008.

# VITA

Aibek Musaev was born in Bishkek, Kyrgyzstan. In 1996 he became a recipient of the Scholarship of the President of the Kyrgyz Republic that provided full funding for $2^1/_2$ years of undergraduate studies in the United States. He received his B.S. and M.S. degrees in Computer Science from Georgia Tech in 1999 and 2000 respectively. After graduation, he joined Siebel Systems, Inc. in San Mateo, CA as an application developer. In 2002 Aibek returned to Kyrgyzstan where he founded a software company Akforta that provided enterprise management solutions to private and public organizations, including State Customs Service and Social Fund. In 2011 he left Akforta to pursue a Ph.D. in Computer Science at Georgia Tech, where he was advised by Prof. Calton Pu. His primary research interests are in applied data mining of big data from multiple sources with the focus on disaster management.